HP OpenView Service Desk

OpenView Operations Integration Administrator's Guide

Software Version: 5.10

For the Windows and UNIX Operating Systems

Manufacturing Part Number: None

Document Release Date: August 2007

Software Release Date: August 2007

Legal Notices

Warranty.

Hewlett-Packard makes no warranty of any kind with regard to this document, 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.

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

OpenView® is a registered U.S. trademark of Hewlett-Packard Company.

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

OSF, OSF/1, OSF/Motif, Motif, and Open Software Foundation are trademarks of the Open Software Foundation in the U.S. and other countries.

Contents

1. Introduction to this Manual

2. Integration Overview

3. OVO for Windows Integration Installation

4. OVO for Windows Integration Configuration

1 Introduction to this Manual

This guide explains the integration between Service Desk and OpenView Operations for Windows® and UNIX®. With the information in this guide you can install, configure and perform the various tasks available with this integration.

This guide is intended for IT administrators who will install and configure the integration, and for users who will perform the integration tasks.

To install and configure the integration you must have knowledge of both the OpenView Operations application and Service Desk.

This guide is organized as follows:

- Chapter 2, "Integration Overview," on page 15 describes the architecture of the OpenView Operations Integration and gives a brief explanation of the integration possibilities.
- Chapter 3, "OVO for Windows Integration Installation," on page 23 describes the installation steps that need to be performed on Service Desk and on your OVO for Windows server.
- Chapter 4, "OVO for Windows Integration Configuration," on page 31 explains how to configure the integration if you are using OVO for Windows.
- Chapter 5, "OVO for UNIX Integration Installation," on page 45 explains how to install the integration if you are using OVO for UNIX with Service Desk.
- Chapter 6, "OVO for UNIX Integration Configuration," on page 51 explains how to configure the integration if you are using OVO for UNIX with Service Desk.
- Chapter 7, "User Tasks," on page 67 provides examples on how to use the different features supplied with this integration.
- Chapter A, "Troubleshooting." contains information on the default locations of installed files and other information that may be helpful in solving problems with the integration.

Related Publications

The manuals in the list below contain information related to this guide.

- *HP OpenView Operations for UNIX: Concepts Guide*
- *HP OpenView Operations: Concepts Guide*
- *HP OpenView Operations: Installation and Administrative Task Guide*
- *HP OpenView Operations for UNIX Developer's Toolkit: Application Integration Guide*

Documentation Updates

This manual's title page contains the following identifying information:

- Version number, which indicates the software version.
- Document release date, which changes each time the document is updated.
- Software release date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition, visit the following URL:

http://ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

NOTE For an overview of the latest information on supported platforms, installation prerequisites, software deliverables, and so on, refer to the Service Desk 5.10 Release Notes.

Table 1-1 Changes to this Document

Chapter	**Changes**
Documentation Updates	Added note on Service Desk 5.10 Release Notes.
4 OVO for Windows Integration Configuration	Modifying Configuration Files on the OVO Management Server, step 4: Changed `mymapping` to `ovowin`. Modify and Deploy Policies on OVO Windows Managed Nodes: Changed information on policies.

Support

Please visit the HP OpenView support web site at:

http://www.hp.com/managementsoftware/support

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

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

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

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

http://www.managementsoftware.hp.com/passport-registration.html

2 Integration Overview

This chapter provides an explanation of the key features and processes in the OpenView Operations integration.

Feature Overview

The HP OpenView Operations integration brings together the functionality of HP OpenView Service Desk (hereafter known as Service Desk) and HP OpenView Operation (hereafter known as OVO) and allows the transfer of information between both. OVO monitors and adjusts your IT environment, while Service Desk manages the processes within this same IT environment. Some of the key features in this integration are described in the following sections.

Bidirectional Event Integration

The OVO Integration provides not only the capability to forward messages to Service Desk but also the ability to acknowledge these messages when the corresponding incident in Service Desk is closed. For example, when an OVO agent registers a problem with a node and a message is created in the OVO management server, the integration will automatically create an incident in Service Desk. This incident is then handled in accordance with your company ITIL processes. When the incident is closed, the corresponding Message in OVO is also closed. Based on the policy/template definitions in OVO, you are able to define the type of messages for each infrastructure element that needs to be forwarded to Service Desk. The flow of information is bidirectional; when the incident is created in Service Desk, an incident ID is passed back to OVO.

Status Update (OVO for Windows)

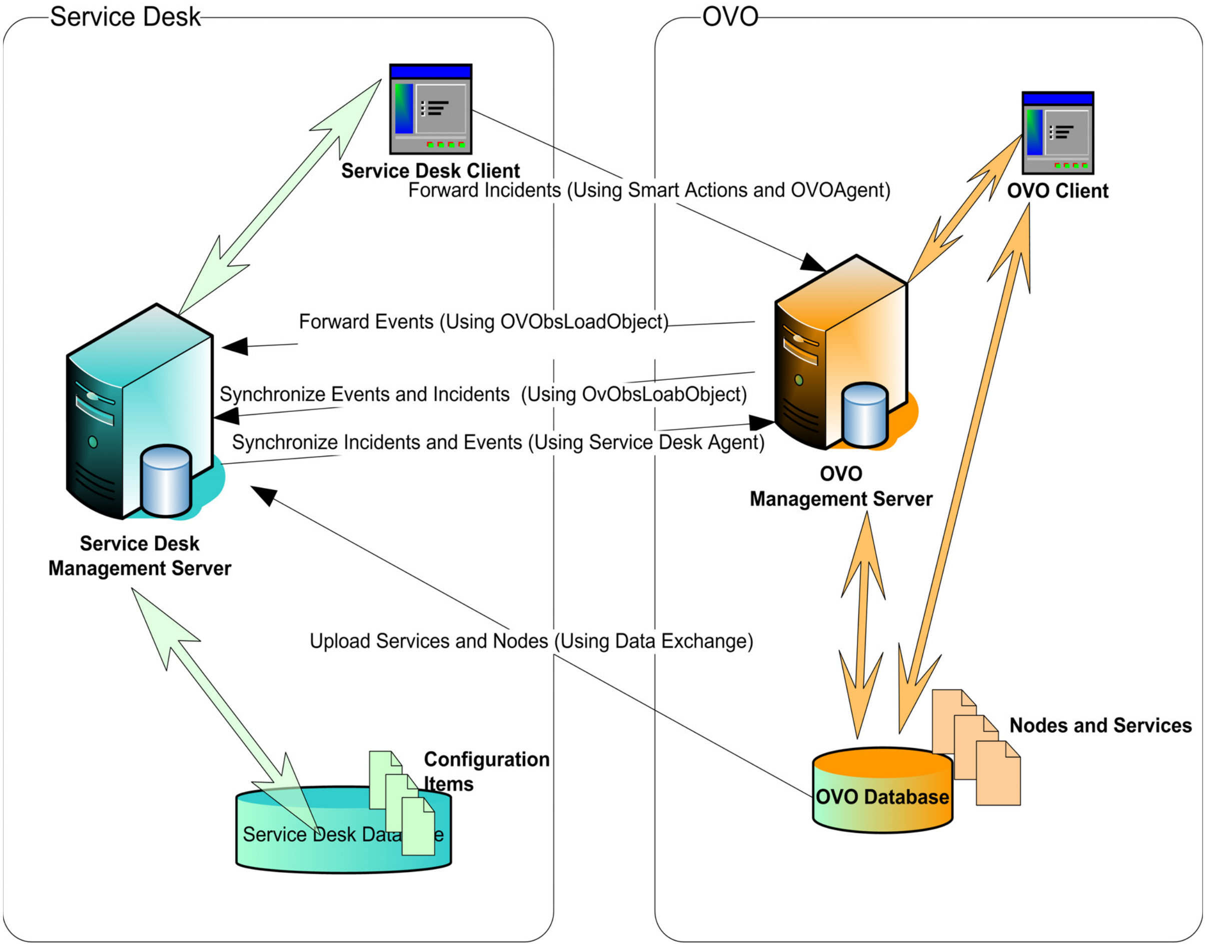
The OVO Integration is designed to forward status updates to an incident in Service Desk when the priority of the message in OVO has changed. This change is automatically reflected in the corresponding Service Desk incident. Similarly, if you add or modify an annotation in OVO for Windows, this information is automatically transferred to Service Desk and the corresponding Service Desk incident is updated. Also, when closing an incident in Service Desk, it is automatically reflected in OVO, providing up-to-date status information for the operators and improving the communication between help desk operators and infrastructure management operators. Changing the severity in Service Desk will result in a severity change of the OVO message. Also, when a status of an incident in Service Desk is changed, an annotation is added to the corresponding OVO message.

Node and Services Upload

Accurate configuration items are fundamental to Service Desk accurately responding to faults in your system. This Integration allows you to upload node and service information from OVO into Service Desk as configuration items. The OVO Integration uses the Service Desk Data Exchange functionality to upload the node information into Service Desk. Services are only uploaded from OVO for Windows.

Integration Architecture and Processes

The OVO Integration comprises of several processes that allow communication of data between Service Desk and OVO. The following diagram shows the process-architecture of the OVO Integration:


Service Desk
OVO
Service Desk Client
OVO Client
Forward Incidents (Using Smart Actions and OVOAgent)
Forward Events (Using OVObsLoadObject)
Synchronize Events and Incidents (Using OvObsLoabObject)
Synchronize Incidents and Events (Using Service Desk Agent)
OVO
Management Server
Service Desk
Management Server
Upload Services and Nodes (Using Data Exchange)
Configuration
Items
Service Desk Database
Nodes and Services
OVO Database

Forward Events from OVO to Service Desk

This Integration automatically forwards OVO events from your OVO environment to Service Desk. See “Configuring Your Environment” on page 53. However, you can also manually forward events from OVO to Service Desk if, for example, you have not configured your environment to forward a particular type of event. See “Manually Forwarding Messages from OVO to Service Desk (OVO for UNIX Only)” on page 71.

You can configure OVO to send all events or specific events to Service Desk. The event information is mapped to a Service Desk incident. The first time an event is sent, an incident is created in Service Desk; Service Desk is then the owner of that event. The import mapping in Service Desk defines which event attributes will be imported into the incident fields.

Uploading Nodes and Services from OVO into Service Desk

This integration allows you to upload services and nodes from your OVO environment to Service Desk. The upload process should be performed on a regular basis so that the CMDB in Service Desk contains the nodes and services that you are managing. See “Exporting and Importing Services and Nodes from OVO into Service Desk” on page 68

Forward an Incident from Service Desk to OVO

This process allows you to forward an incident to OVO as an event (also known as a message), if there is not already a corresponding event in OVO for that incident.

Send Message Annotations from Service Desk to OVO

A message is sent to OVO stating that the event forwarded to Service Desk has been registered as an incident (with the corresponding incident number). Annotating a message in OVO is similar to adding a note of explanation to a business contract. The annotation is a short summation of the important points and can be used as a reference the next time you receive the same message. Message annotations are normally used to provide information on:

- action performed to resolve the problem;

- name of the user who started the action;
- status information for the action performed;
- start and finish time of the action;
- any pre- or post action information which is relevant.

Database rules are available with this integration for sending an annotation to OVO whenever an incident is created as a result of an event being sent from OVO to Service Desk. The database rule can also be configured to send an annotation to the OVO application whenever the status of that incident changes. Database rules can be turned on/off and modified in Service Desk. Service Desk agents on the OVO servers send commands generated from a database rule in Service Desk.

Once the integration is installed and configured, an annotation is sent whenever Service Desk receives an incident from OVO. Optionally, whenever a Service Desk user updates the incident's information, an annotation is also sent to OVO.

Send Acknowledgment Messages from Service Desk to OVO

Messages are sent to OVO when the Status or Severity of an incident is changed. When the status of an incident is set to Closed, this is referred to as acknowledging an event in OVO.

Database rules are available with this integration for acknowledging messages/events from OVO whenever an incident that was created as a result of an OVO event is closed in Service Desk. Agents on the OVO server send commands generated by a database rule in Service Desk.

Once the integration is installed and configured, whenever an incident from OVO is closed in Service Desk, an acknowledgement is sent automatically.

Monitor Service Desk Processes and Error Log Files in OVO for Windows

You can instruct OVO for Windows to monitor Service Desk log files.

OVO policies can be used to monitor the log files for the Service Desk agent and the Service Desk management server. This provides you with a means for monitoring the integration continuously. The policies can be used to match specific log file lines and assign variables out of the intercepted lines to use for pattern matching.

Once the policies are deployed, detected errors are sent to the message browser automatically. Instructional text is provided with the error messages to help solve the detected problems. Use the policies `Monitor_LoadObject_logfile` and `Monitor_LoadObject_error-logfile` to monitor the OVO management server. Use the policy `Monitor_Service_Desk_logfile` for monitoring the Service Desk management server.

For every log file, a rule has to be defined in the policy, according to the following convention (the logfile policies to be created are basically a copy of the existing one):

The naming convention is:

```
<filename>.X.Y.<locale>
```

- `filename` defaults to system.
- `X` refers to the number of files with the same name. `System.1` means that there is a `System.0`.

  Every OpenView process that uses this form of logging will create a new log file. If the file being created is already in use, the `.X` is appended to create a unique file name.
- `Y` is the roll-over counter. All `System.0.Y` files belong together and are used for logging information of the same process.
- `locale` can be used to enable translated versions of logging messages for different languages. The availability of these localized messages depends on the presence of localized versions on the system of the messages being logged. These translated messages should typically be provided by the product.

Suppress Node-down Messages in OVO for UNIX

When Service Desk schedules an outage (for example, a node is shutdown for a particular time interval), OVO automatically suppresses messages related to that node.

You can prevent Service Desk from receiving OVO for UNIX messages by letting OVO suppress the messages when an outage is planned. This is done by using `mkoutage` to retrieve data from Service Desk and pass this information to OVO. Two types of outages are recognized: periodic (or recurring outages), and occasional (or incidental outages). See “Modify Database Rules Related to the Outage Functionality” on page 55.

Reflect OVO for Windows updates in Service Desk

If an incident is created in Service Desk based on a message from OVO and the message is updated in OVO (new annotation is sent from OVO to Service Desk or severity changes are sent from OVO to Service Desk), then the changes are reflected in Service Desk.

3 OVO for Windows Integration Installation

This chapter describes how to install the OVO integration on a Windows platform.

About the OVO Integration (Windows) Installation

The OVO Integration installs the software and utilities that are needed to perform the integration: Object Loader, Service Desk Agent, and Data Exchange.

Install OVO Integration on the OVO Management Server

Installation Prerequisites

- Windows installed with the latest patches and running
- OVO for Windows installed and running
- Service Desk management server installed and running
- The object loader tool installed and running (for instructions, see *Service Desk Installation Guide*)
- HTTP protocol enabled on the Service Desk management server. To enable the HTTP protocol:

  1. Log on to the system on which the Service Desk management server is installed.
  2. Depending on the operating system, run one of the following commands to launch the Server Configuration program:

     — UNIX: `/opt/OV/bin/OvObsServerSettingsEditor`.

     — Windows: **Start**→**Programs**→**HP OpenView**→**Server Settings**.

  3. Navigate to the Protocols tab.
  4. Select HTTP from the **Protocol** drop-down list.
  5. Select the **Enable this protocol** check box.
  6. From a command prompt, use the `ovc` command to restart the ovobs process:

     ```
     ovc -restart ovobs
     ```

- Service Desk client installed and running on the OVO management server

Before you Start the Install Procedure

- Read the contents of Chapter 2 “Preparing to Install HP OpenView Service Desk 5.0” of the HP OpenView Service Desk Installation Guide, including the section “HP OpenView Installer – Overview” – this contains a description of the HP OpenView Installer and associated screens.
- Ensure that you have the appropriate permissions required to install the software. You need Administrator privileges to install this software.

Install Procedure

NOTE If you perform an upgrade installation, you need to perform additional tasks after completing this procedure. For further information, see “Upgrading from Previous OVO Integration (Windows) Versions” on page 28.

1. Log on to the host system on which the OVO for Windows management server is installed.
2. Locate the execution file:

   ```
   <UnpackDir>\ovowintegration_05.xx.yyy_setup.exe
   ```

3. Run `ovowintegration_05.xx.yyy_setup.exe` to start the installation. The Introduction window of OpenView Installer opens.
4. The installation now follows the standard OpenView Installer process as described in Chapter 2 of the HP OpenView Service Desk Installation Guide.
5. Log in to the OVO console and deploy the OVO agent on the Service Desk client machines that need to send events to OVO (see the OVO user documentation for installation instructions). The OVO agent is required by the OVO integration to generate an OVO message from Service Desk.
6. Deploy `opcmsg` (from the OVO console):

   a. In the OVO management console, open **Policies grouped by type**.

   b. Open the **Open Message Interface** policy group.

c. Select the **opcmsg** policy, right-click and then select **All Tasks**, then **Deploy on**.

d. Select all of the Service Desk client nodes that you want to be able to manually create OVO messages from and click **OK**.

Opcmsg is used to create OVO messages from Service Desk. You need to deploy this policy to all Service Desk clients that you want to be able to perform this function from.

For additional information refer to the Online Help in your *OpenView Operations for Windows* application.

7. Disable the monitor logfile policies until they are configured. Please see “Modify and Deploy Policies on OVO Windows Managed Nodes” on page 36 for instructions on configuring the logfile policies.
8. See Chapter 4, “OVO for Windows Integration Configuration,” on page 31 for further instructions on how to complete the setup of this integration.

WARNING **To support multiple OVO management servers you need to install the OVO Integration on each server.**

NOTE Once the installation steps are completed, check if the Service Desk agent is running (using the command `ovc -status`); `ovobsag` should be listed as a running service. If the service is not running, type the following command: `ovc -start ovobsag`.

Upgrading from Previous OVO Integration (Windows) Versions

If you upgrade an OVO integration from Service Desk 5.00 to Service Desk 5.10, you need to upload the integration configuration data into Service Desk again. See “Uploading Configuration Data into Service Desk” on page 33.

Before you start the upgrade, you should rename the following files, for example, with the extension .old, in the <DataDir>\Data\Datafiles\data_exchange\config directory:

- OvSdOvowCi.conf
- OvSdOvowServices.conf
- OvSdOvowCi2003.conf
- OvSdOvowServices2003.conf

You should do this to signify that they are no longer used, and you may need to re-enter any customer-specific information contained in these files in the new OvSdOvow.conf file. Subsequently you can remove the files, but this is not mandatory.

After the upgrade a new Import Mapping ovowin is created in Service Desk. As a result, any customizations you have made to the old Service Desk Import Mappings must be reapplied to the new Import Mapping ovowin. The old Import Mappings are:

- ovowindowsci (import mapping for configuration items)
- ovowindowsservices (import mapping for services)
- ovowindowsci2003 (import mapping for configuration items on Windows 2003)
- ovowindowsservices2003 (import mapping for services on Windows 2003)
- ovowindows (import mapping for incidents)

Subsequently you can remove the old Import Mappings, but this is not mandatory.

In OVO for Windows a new policy, `Monitor LoadObject logfile`, has been created. You can remove the old policy `Monitor sd_event logfile`, but this is not mandatory.

4 OVO for Windows Integration Configuration

This chapter describes the integration programs that allow communication between Service Desk and OVO.

Overview

This section explains the configuration steps that must be done in Service Desk. The tasks are listed below:

The following will be created automatically (after uploading the Configuration Exchange file):

- OVO Integration account
- Database rules
- OVO Integration Role
- Import Mappings
- Templates
- Smart Action

The configuration tasks are:

- Modify the `OvObsLoadObject.conf` file.
- Configure the Integration Account.
- Deploy the forwarding policy and change event policy.
- Edit logfile monitoring policies.
- Configuring the OvServiceExport tool for viewing OVO services in a browser.

Uploading Configuration Data into Service Desk

The OVO Integration installs configuration files that contain the configuration data needed by the integration. The configuration exchange file contains the following data:

- Integration Account
- Integration Role
- Database Rules
- Import Mappings
- Object templates
- Smart action

To upload this configuration data into Service Desk:

1. In the OpenView Console, select **Configuration Exchange** from the **File** menu, and then select **Import**.

   The Configuration Exchange Import Wizard opens.

2. Follow the instructions and import the following file:

   ```
   <install_dir>\Program Files\HP
   OpenView\Data\Datafiles\data_exchange\xml\OvSdOvow.xml.
   ```

   See “Importing Configuration Settings” in the OpenView online help.

Modify Account Information in Service Desk

The integration account information in Service Desk (imported as part of the configuration exchange import) needs to be modified.

1. Log on to the OpenView Console.
2. The host name must be modified to reference the OVO management server. Select the **Users and Security** workspace from the OV Configuration workspace group. Select **Access > Accounts > OVO for Windows**. In the OVO for Windows - Account dialog on the General tab, change **Host** to the host name of the OVO management server. On the Role tab, select the Integration account role.

   If you have multiple OVO management servers, create an integration account for each additional OVO management server.

   For single and multiple OVO management server environments, it is good practice to name each account `OVOW_<OVO_server_name>`, where `<OVO_server_name>` is the OVO management server name. Each account name must start with `OVOW`.
3. Change the integration account password - This is an optional setting, but for security reasons it is advised to change the OVO for Windows account password in Service Desk. Please refer to the online help (Password Settings) for information on how to change your default password. The default password is `Password4OpenV!ew`.

Modifying Configuration Files on the OVO Management Server

In order to insert information into Service Desk from OVO (event forwarding), you need to modify the `OvObsLoadObject.conf` configuration file. This file was installed during the OVO for Windows integration installation (see “Install OVO Integration on the OVO Management Server” on page 25)

To modify this file:

1. Open `<DataDir>\conf\obs\loadobject\OvObsLoadObject.conf` in a text editor.
2. In the `ACCOUNT=myusername/mypassword` entry, replace `mypassword` with the previously defined integration account username and password. The account name must start with OVOW because Service Desk database rules contain conditions that are dependent on this naming convention.
3. Replace `SERVER=localhost` with the host name of the Service Desk management server.
4. In the `MAPPING=mymapping` entry, replace `mymapping` with `ovowin`.

Modify and Deploy Policies on OVO Windows Managed Nodes

Policies provided by this integration allow you to monitor Service Desk log files. The policies are:

- `Monitor_LoadObject_logfile.policy`

  This policy allows you to monitor warnings in the LoadObject log file that occurred during the LoadObject process.

  Once uploaded to OVO for Windows, this policy is visible as Monitor LoadObject logfile.

- `Monitor_LoadObject_error-logfile.policy`

  This policy allows you to monitor errors in the LoadObject error log file that occurred during the LoadObject process.

  Once uploaded to OVO for Windows, this policy is visible as `Monitor LoadObject error-logfile`.

- `Monitor_Service_Desk_logfile.policy`

  This policy allows you to monitor java exceptions and warnings in a Service Desk log file that occurred on the SD Management Server and the SD Agent.

  Once uploaded to OVO for Windows, this policy is visible as `Monitor Service_Desk_logfile`.

In order to have the policies monitor Service Desk log files (`<install_dir>\Data\Log`), the Service Desk Logfile Monitoring policy must be edited through the OVO for Windows console in the Service Desk Policy Group. For every Service Desk log file, a separate rule needs to be created. See “Monitor Service Desk Processes and Error Log Files in OVO for Windows” on page 20.

You need to deploy the forwarding policies on the OVO for Windows management server. You may also need to modify the conditions of the forwarding policies to avoid the forwarding of all events regardless of the circumstances. Refer to the OVO for Windows documentation for instructions on how to deploy and filter policies in an OVO for Windows environment.

Viewing OVO Services in a HTML Browser

To view OVO for Windows services in a browser you need to export the data from the OVO database using the OvServiceExport tool. This tool exports the data from the OVO for Windows database in either HTML or XML format for viewing in a browser.

OvServiceExport Tool

OvServiceExport is a command line tool that enables you to export information about an OVO for Windows service.

OvServiceExport.exe accepts the following optional parameters:

```
OvServiceExport.exe -server <server name> -service <service
name> [-format XML -file <file name>] ¦ [-format HTML -cgi
<cgi root> -file <file name>]
```

Table 4-1 describes the parameters:

Table 4-1 OvServiceExport Command Line Parameters

Parameter	Description
-server	<server name> is the name of the OVO management server that holds the service information. Default value: name of the local machine
-service	<service name> is the name (not the caption) of the service of which you want to export information. Default value: `Root_Services`

Table 4-1 **OvServiceExport Command Line Parameters (Continued)**

Parameter	Description
-format	Choose between XML and HTML: • XML The information is generated in XML format. This XML stream contains service information about the service <service name>, all association information about its dependencies and components as well as service information for each of these dependencies and components. The XML stream follows the rules of `service.dtd`, the same DTD that OpenView Operations for Unix uses. • HTML Use this option, for example, if you want to use `OvServiceExport.exe` together with CGI. The information is formatted in a special form of HTML. The HTML page includes information about the service <service name> as well as the information of its child and dependent services. Default value: XML
-file	<file name> is the name of the output file. If no file name is specified, the information is written to standard output. Default value: none
-cgi	If you use `OvServiceExport.exe` together with CGI, <cgi root> specifies the directory where the executable is located on the web server. Default value: cgi-bin
-refresh	<refresh time> is the time in seconds after which the HTML page on your browser is updated by a new CGI request. This option is only applicable if you use CGI. Default value: 60

Generating XML Output

If you want to generate an XML stream displayed on standard output, run the following command:

```
OvServiceExport.exe -format xml
```

You can add other parameters (see Table 4-1 on page 37) from the command line.

Another example:

```
OvServiceExport.exe -format xml -server VPWServer -service
VP_SM:Server -file test.xml
```

This example generates a file called `test.xml` with the service information of the service `VP_SM:Server` from the management Server `VPWServer`.

Using OvServiceExport.exe with CGI

This section describes how to use `OvServiceExport.exe` with CGI.

Installing OvServiceExport on a Web Server

It is important that the `OvServiceExport.exe` runs under an account that has the rights to retrieve data from OVO for Windows, as explained below.

Apache Web Server If you use an Apache web server, you need to copy the `OvServiceExport.exe` file to a directory from where Apache is able to run CGI scripts. The Apache default directory is `cgi-bin`.

If you use a non-default directory, use the `-cgi <cgi root>` parameter of `OvServiceExport.exe` to specify it.

Normally, Apache runs the executable file under the system account, so you should not have any security issues.

The `htdocs` directory must contain an `OvServiceExport` directory. The `OvServiceExport` directory must contain the two images `VPW.gif` and `VPWSvcIco.gif`.

Microsoft Internet Information Services 6.0

1. Create folder `C:\Inetpub\cgi-bin` on the OVO management server.

2. Navigate to the `<install_dir>\Program Files\HP OpenView\misc\sd\ovow` directory.
3. Extract the `OvSdServiceExport.zip` file to a directory of your choice.
4. Copy the `OvServiceExport.exe` file to the `C:\Inetpub\cgi-bin` directory.
5. Copy the `OvServiceExport` folder to `C:\Inetpub\wwwroot`. Verify that this folder contains the `VPWSvcIco.gif` and `VPW.gif` files.
6. Open the IIS Manager (IISM).
7. In IISM under `Web Sites->Default Web Site`, create the `cgi-bin` directory as an IIS virtual directory. Use `cgi-bin` as the alias and set `C:\Inetpub\cgi-bin` as the path.
8. Right-click **OvServiceExport.exe** in IISM window and select **Properties**.
9. Open the File Security tab and click **Edit** in the **Authentication and access control** section. In the opened dialog, do the following:
   - Select the **Anonymous access** check box and click **Browse**
   - Specify a user account and password that has the rights to access OVO Services. Typically that would be the `<DOMAIN>\HP-OVE-User` account.
10. In the folder tree, right-click **cgi-bin** and select **Properties**. Select the Virtual Directory tab and verify that **Execute permissions** is set to **Scripts and Executables**. If not, manually select it. When prompted to override inheritances, select all. Accept these settings and return to the ISSM console.
11. In the folder tree, right-click **Default Web Site** and select **Properties**. Select the Home Directory tab and verify that **Execute permissions** is set to **Scripts and Executables**. If not, manually select it. When prompted to override inheritances, select all. Accept these settings and return to the ISSM console.
12. In the folder tree, right-click **Web Service Extensions** and select **Add new Web service extension**:
    - Add extension name.
    - Add `C:\Inetpub\cgi-bin\OvExportService.exe`.

- Select the **Set extension status to allowed** check box.

13. Exit the ISSM console.
14. To verify that the web service viewer is working, start Internet Explorer and type the following URL: <http://<web_server_name>/cgi-bin/OvServiceExport.exe?-format +html>

Using the HTML Service Viewer of OvServiceExport

Now everything should be ready to use `OvServiceExport.exe` with CGI. The URL you need to specify in a web browser to call `OvServiceExport.exe` follows the following format:

```
http://<web server name>/<cgi
root>/OvServiceExport.exe?-format+html+ Paramneter2+<Value
of Parameter2>+Parameter3+<Value of Parameter3>+...
```

Example 1 Suppose you installed `OvServiceExport.exe` in the `cgi-bin` directory of your Apache web server on the machine `webserver.mydomain.com`. `webserver.mydomain.com` is also the OVO management server.

The easiest call would be:

```
http://webserver.mydomain.com/cgi-bin/OvServiceExport.exe?-f
ormat+html
```

This URL would return the status of the Root_Services on `webserver.mydomain.com`.

Example 2 Suppose you installed `OvServiceExport.exe` in the directory `MyDirectory` in the `scripts` directory of your IIS 6.0 configuration on the server `webserver.mydomain.com`. `VPW.mydomain.com` is the OVO management server.

Consider the URL:

```
http://webserver.mydomain.com/scripts/MyDirectory/OvServiceE
xport.exe?-format
+html+-cgi+scripts/MyDirectory+-server+VPW+-service+VP_SM:
Server+-refresh+120
```

This URL would return the status of the service `VP_SM:Server` from the management server `VPW.mydomain.com`. The HTML page would be refreshed every two minutes (120 seconds).

Handling of Non-ASCII Characters

OvServiceExport is able to cope with different languages. Therefore, within the generated XML/HTML output streams, non ASCII characters are replaced using the following HTML syntax:

"&#x Hexadecimal value of Unicode character".

For example, &#xDF represents the character ß.

NOTE If you are using OvServerExport together with CGI, you cannot use a service with non-ASCII characters in its ID (the Caption field does not have this restriction). This would generate invalid CGI calls.

There is no similar restriction if you are writing to files.

Exporting Nodes and Services from OVO for Windows into Service Desk

To import OVO for Windows nodes and services into Service Desk, you must first export the nodes and service information from OVO into configuration items in Service Desk.

Editing Configuration Files for Importing Nodes and Services Into Service Desk on Windows

To import nodes and services into Service Desk from OVO for Windows, you have to edit the configuration file. This has to be done on every Service Desk client that will perform the OVO for Windows imports. This file is installed during the OVO for Windows integration installation (see “Install OVO Integration on the OVO Management Server” on page 25).

1. Navigate to `<DataDir>\Datafiles\data_exchange\config`.
2. Modify the file `OvSdOvow.conf` under the JDBC section. Locate the `Server` entry and specify the OVO server as the value, for example `//MYOVOSERVER`.

NOTE On a stand-alone client you need to copy the SQL Server JDBC driver from a Service Desk management server to your Service Desk client. The SQL server JDBC driver is `<InstallDir>\nonOV\obs\Opta.jar`. Copy this file to `<InstallDir>\nonOV\obs` on your client.

5 OVO for UNIX Integration Installation

This chapter describes how to install the OVO integration on a UNIX platform.

About the OVO Integration (UNIX) Installation

The OVO Integration (UNIX) installs all the software and the utilities that are needed by the integration: OvObsLoadObject, OvObsAgent, and Data Exchange.

Install OVO Integration on the OVO Management Server

Installation Prerequisites

- HP-UX installed with the latest patches and running
- OVO for UNIX installed and running
- Service Desk management server installed and running
- HTTP protocol enabled in Server Settings on the Service Desk management server. To enable the HTTP protocol:

    1. Log on to the system on which the Service Desk management server is installed.
    2. Depending on the operating system, run one of the following commands to launch the Server Configuration program:

        — UNIX: `/opt/OV/bin/OvObsServerSettingsEditor`.

        — Windows: **Start→Programs→HP OpenView→Server Settings**.

    3. Navigate to the Protocols tab.
    4. Select HTTP from the **Protocol** drop-down list.
    5. Select the **Enable this protocol** check box.
    6. From a command prompt, use the `ovc` command to restart the ovobs process:

        ```
        ovc -restart ovobs
        ```

- Service Desk client installed and running (it is not mandatory to install the Service Desk client on the OVO management server)

Before you Start the Install Procedure

- Read the contents of Chapter 2 “Preparing to Install HP OpenView Service Desk 5.0” of the HP OpenView Service Desk Installation Guide, including the section “HP OpenView Installer – Overview” – this contains a description of the HP OpenView Installer and associated screens.

- Ensure that you have the appropriate permissions required to install the software. You need root access to install this software.

Install Procedure

NOTE If you perform an upgrade installation, additional tasks will be performed during the configuration phase. For further information, see "Upgrading from Previous OVO Integration (UNIX) Versions" on page 57.

1. Locate the execution file:

   `<UnPackDir>/ovouintegration_5.xxx.yyy_setup.bin` (where xxx is the version number, and yyy is the build number).

2. Run the execution file to start the installation. The **Introduction** window of HP OpenView Installer opens.

3. The installation now follows the standard HP OpenView Installer process as described in Chapter 2 of the HP OpenView Service Desk Installation Guide.

NOTE The OVO for UNIX integration install procedure installs all the necessary software packages needed to run this integration.

4. Login to the OVO console and deploy the OpenView for Operations agent on the Service Desk client machine; see the OVO user documentation for installation instructions. This step only needs to be performed for Service Desk clients that need to send events to OVO. The OVO agent is required by the OVO integration to forward an incident from Service Desk to the OVO management server.

5. To support multiple OVO management servers you need to install the OVO Integration on each OVO management server.

6. See Chapter 6, "OVO for UNIX Integration Configuration," on page 51 for further instructions on how to complete the setup of this integration.

NOTE

Once the installation steps are completed, check if the Service Desk agent is running on the OVO management server (using the command `ovc -status`); `ovobsag` should be listed as a running service. If the service is not running, type the following command: `ovc -start ovobsag`.

The installation of the integration software on the OVO management server stops the OVO agent that runs on the OVO management server. You need to restart the OVO agent manually using the command `/opt/OV/bin/opcagt -start`.

6 OVO for UNIX Integration Configuration

This chapter describes the configuration tasks to be performed in Service Desk and on the OVO for UNIX management server.

Overview

The OVO integration configuration consists of automated and manual tasks:

- Upload configuration data into OVO Operations - See “Running the Configuration Script” on page 53
- Creation of configuration files for each OVO management server (automatic) - See “Running the Configuration Script” on page 53
- Preparation of configuration data for uploading into Service Desk (automatic) - See “Running the Configuration Script” on page 53
- Upload the configuration data into Service Desk (manual) - See “Uploading the Configuration Data into Service Desk” on page 53
- Modify database rules related to the outage functionality (manual)
- Change default passwords (manual, optional)
- Perform Configuration Tasks in OVO - See “Configuration Tasks in OVO” on page 56
- Deploy the Oracle JDBC Driver - See “Deploy the Oracle JDBC Driver” on page 57.

Configuring Your Environment

This section describes the main configuration steps that must be performed before the integration can begin the bidirectional transfer of information between OVO and Service Desk.

Running the Configuration Script

This section describes the tasks that you need to perform in order to properly configure your system for this integration. To start the configuration process, run the following command on the OVO management server: `/opt/OV/bin/ovsdovosetup.sh`

Follow the instructions shown on screen. Once the script has completed, perform the tasks described in the following sections.

NOTE

If you answer **yes** to the question `Do you want to upload the configuration data for the integration with Service Desk into OpenView Operations?` then you must restart the OVO processes when the configuration script has finished. Use the commands `/opt/OV/bin/OpC/opcsv -stop` and `/opt/OV/bin/OpC/opcsv -start` to stop and start the OVO server processes.

Uploading the Configuration Data into Service Desk

Automatic or Manual Configuration

If you perform the installation and configuration of the OVO for UNIX integration on an OVO management server on which a Service Desk management server is also installed, the upload task is performed automatically if you answer **yes** to the question `Do you want to upload the configuration data for the integration with OpenView Operations into Service Desk?` In this case you do not need to perform the manual configuration described below.

To manually upload configuration data into Service Desk, perform the following tasks:

Upload the Service Desk configuration data using the Service Desk client. After running the `ovvsdovosetup.sh` script (this file prepares the configuration data file for uploading), copy the file

```
/var/opt/OV/conf/obs/data_exchange/ovsdintegrationsdconfig
.xml
```

to a Service Desk client machine. Start the Service Desk client and select **File→Configuration Exchange→Import**. The **Configuration-Exchange Import Wizard** appears. Click **Next** on the first page of the import wizard and click **Add**, and browse to the directory containing `ovsdintegrationsdconfig.xml`, select this file, and click **Start import**. After importing the configuration data, exit the client.

NOTE After uploading the configuration data into SD either automatically or manually you must restart the SD server processes on the SD management server. Use the commands `ovc -stop` and `ovc -start` to stop and start the SD server processes.

If you have multiple OVO management servers, you need to run the `ovsdovosetup.sh` script for each server.

You only need to upload the Service Desk configuration data once. Therefore, do not upload the configuration data into Service Desk if you have already performed an upload during the configuration of the first OVO management server.

You need to create an integration account for each additional OVO management server. Check that the following information is supplied under Accounts in the OpenView Console:

- The Account name starts with `OVOU`.

  It is good practice to include the name of the OVO management server in the name of the account. For example, if the hostname of your OVO management server is `merlin`, name the account `OVOU_merlin`.

- In **Host**, type the hostname of the OVO management server.
- Set **Account Type** to Integration.

  Under the **Roles** tab, select the **OVO Integration** role

Modify Database Rules Related to the Outage Functionality

You need to modify four database rules to facilitate outage work. These rules are:

- OVO occasional outage from work order - For work orders
- OVO periodic outage from CI - For configuration items
- OVO periodic outage from schedule - For schedules
- OVO periodic outage from schedule element- For schedule elements

To modify database rules:

1. Log into the HP OpenView Console using a role that includes administrator privileges and select the work-space category **HP OpenView Configuration**. Detailed information on database rules is available in the online help.
2. Navigate to the **Actions & Rules** work space. In the **Actions & Rules** tree, select **Rules**.
3. In the **Rules** tree, select **Database Rules**. First, select the object (Workorder, Schedule, Configuration Item, and Schedule Element) and then select the rule you need to modify:
    - OVO occasional outage from work order
    - OVO periodic outage from CI
    - OVO periodic outage from schedule
    - OVO periodic outage from schedule element
4. Navigate to the **Command Exec** action page of the database rules wizard and click **Edit** for the action you want to modify (under “**What actions do you want to be performed?**”).
5. The **Command Exec Action** window displays. Enter the hostname of the OVO management server (in place of `myovoserver.mydomain.com`) in the **Host** section.

NOTE If you have multiple OVO management servers, you need to create additional Command Exec Actions in Service Desk for each server. Enter the hostname for each management server in your system.

Change Default Passwords (manual, optional)

This is an optional setting, but for security reasons it is advised to change the OVO for UNIX and OVO for UNIX Outage account passwords in Service Desk. Please refer to the online help (Password Settings) for information on how to change your default passwords. The default password is `Password4OpenV!ew`.

Once the passwords are changed in Service Desk, you have to duplicate these changes in the Integration configuration files on the OVO management server. Open the configuration files in a text editor and locate the password entry. The files are:

- `ovsdmkoutage.conf`: located in `var/opt/OV/conf/sd`
- `ovsdevent.conf`: located in `var/opt/OV/conf/obs/loadobject`

NOTE If you run the `ovsdovosetup.sh` script more than once after you changed the passwords, you must repeat the password change as described in this section. The account passwords that you have already specified above are replaced by default passwords every time the script is executed.

The OVO for UNIX Outage account consumes one named user license. An integration account cannot be used.

Configuration Tasks in OVO

Log in to the OVO console and add the responsibility of the Service Desk message group to the appropriate operators.

Log in to the OVO console and configure OVO message source templates by switching on the **Forward to Trouble Ticket** for messages you want forwarded to Service Desk. Assign and install/update the modified templates to the managed node(s). See the OVO for UNIX user documentation for detailed instructions.

NOTE

The Smart Action **Send incident to OV Operations** in the Service Desk console uses the `opcmsg` command to send a message from Service Desk to OVO.

If you configure the `opcmsg` template to forward all messages to the Trouble Ticket interface, a second incident will be created for every incident that is sent from Service Desk to OVO. To prevent duplication of incidents, you should configure an additional condition of the `opcmsg` template to not forward messages to the Trouble Ticket Interface for messages that contain ServiceDesk in the application parameter.

Deploy the Oracle JDBC Driver

Copy the Oracle JDBC driver `ojdbc14.jar` to the directory `/opt/OV/java/drivers` on the OVO management server; this file is located on the Service Desk management server in the `nonOV/obs` subdirectory of the installation directory. The Oracle JDBC driver is required for accessing the OVO Oracle database.

Make sure the jar file has the correct ownership and permissions. The file must be owned by bin and must be in the bin group. Issue the following command from a command line to set the correct ownership:

```
chown bin:bin ojdbc14.jar
```

The permissions of the jar file must be read-only for everyone. Issue the following command from a command line to set the correct permissions:

```
chmod 444 ojdbc14.jar
```

NOTE

The Oracle JDBC driver must be deployed on each OVO management server.

Upgrading from Previous OVO Integration (UNIX) Versions

The Configuration Script `/opt/OV/bin/ovsdovosetup.sh` detects if an upgrade is needed and takes appropriate actions.

The script removes obsolete files from the system.

Existing configuration files are overwritten. However a backup is created of every configuration file. The backup files are located in the same directory as the original files. The file names contain an additional number. Up to 10 revisions of the configuration files are saved. If you made manual changes to the configuration files you must manually re-apply the changes.

If the upload of the configuration data into Service Desk is performed automatically (because a Service Desk management server is installed on the OVO management server), existing configuration data in Service Desk will *not* be overwritten. Therefore you must make the changes to the configuration data manually. To do this, you modify the database rule `Send annotation to OVO for UNIX`, as described in Manually Modifying Configuration Data.

If the upload is not performed automatically, you must perform the upload manually as described in “Uploading the Configuration Data into Service Desk” on page 53. However, in this case, make sure you select "Overwrite existing items" in the Configuration Exchange Import Wizard. Note that *this will overwrite any changes you made to the configuration data in Service Desk*. If you do not want to overwrite the existing configuration data, you must modify the configuration data manually. To do this, modify the database rule `Send annotation to OVO for UNIX`, as described in Manually Modifying Configuration Data.

Manually Modifying Configuration Data

Using a Service Desk client, go to **Actions & Rules→Rules→Database Rules**. Select the rule `Send annotation to OVO for UNIX`. Modify the action of this rule. Replace the command line

```
/opt/OV/lbin/sd/opcannoadd
```

with

```
/opt/OV/lbin/sd/ovsdupdatemessage.sh
```

Changes from Previous Versions

Besides some minor changes and bug fixes, the most import changes are:

- Upload of the configuration data into Service Desk is performed automatically if a Service Desk management server is installed on the OVO management server.
- With previous versions, the operator credentials to access the OVO database is hard-coded in the integration scripts. With this version, the credentials are stored in the configuration file `/var/opt/OV/conf/sd/ovsdovo.conf`.
- With previous versions, OVO messages that are forwarded to Service Desk are always owned by the operator called SD. With this version, owning the messages is optional and can be configured with the OWN entry in the `/var/opt/OV/conf/sd/ovsdovo.conf` file.
- With previous versions, the owning of the messages is performed by the script called by OVO's Trouble Ticket interface. With this version, the owning of the messages is performed by the database rule "Send annotation to OVO for UNIX". Consequently, only when the creation of a Service Desk incident is successful is the OVO message owned by SD.
- Improved error handling. For example, when the operator credentials in OVO do not match the credentials in the `/var/opt/OV/conf/sd/ovsdovo.conf` file, an appropriate error message is written to the `/var/opt/OV/log/ovsdevent_error.log` file.

Details of the Configuration Data

Configuration Data Uploaded into Service Desk

The configuration data is stored in the ovsdintegrationsdconfig.xml file and is uploaded into Service Desk during the configuration phase either automatically or manually.

The configuration data for running the OVO for UNIX integration consists of the following:

- Role OVO Integration

  The role grants access to configuration items, incidents and services.

  The role is given to the OVO for UNIX account.

- Role OVO Outage

  The role grants access to application accounts, configuration items, schedules, schedule elements and work orders.

  The role is given to the OVO for UNIX outage account.

- Account OVO for UNIX

  The account has the role OVO Integration. It is of account type "Integration". The account is used by the "exchange nodes" and "event" interfaces to create configuration items and incidents. The account name must start with OVOU because database rule conditions are defined based on this naming convention. The Host field of the account is used to determine on which OVO management server messages must be owned, annotations to messages must be added and messages must be acknowledged.

  The login name and password must match the login name and password in the ovsdevent.conf file.

- Account OVO for UNIX outage

  The account has the OVO Outage role. It is of account type "Named user". The account is used by the outage interface to log on to Service Desk to retrieve outage information needed for creating an OVO outage file.

The login name and password must match the login name and password in the `ovsdmkoutage.conf` file.

- Incident event type `OVO event`

  Every incident created by the OVO integration will be of type OVO event. It is used by the ovounix import mapping. The database rules Send annotation to OVO for UNIX and Send acknowledgement to OVO for UNIX use this event type in their conditions.

- Configuration item template `OVO Node`

  The template is used for creating a configuration item by the OVO integration. It is used by the ovounix import mapping. Default values are defined for the fields Unique (Yes), Category (System), Remark (Node imported from OVO), Blocked (No), Status (Installed) and Max. Installations (1).

- Incident template `OVO Event`

  The template is used for creating an incident by the OVO integration. It is used by the ovounix import mapping. Default values are defined for the fields State (Registered), Description (OVO generated event), Source Type (OVO event), Impact (None) and Wait for Work Order Completion (No).

- Import mapping `ovounix`

  The import mapping contains two external entities: ovo_node and ovo_event. The mappings are used to create configuration items and incidents respectively. The templates used are "OVO Node" and "OVO Event" respectively.

  For ovo_node, the external attributes SEARCHCODE, NODE_NAME and IP_ADDRESS are mapped to the internal attributes Search Code, Name and IP Address. The IP address is used as primary key to uniquely identify the configuration item.

  For ovo_event, the external attributes solution, severity, information, impact, event_id, description and ci are mapped to the internal attributes Solution, Severity, Information, Impact, Source ID, Description and Configuration Item. The Source ID is used as primary key to uniquely identify the incident

- Database rule for incident `Send annotation to OVO for UNIX`

This database rule is used for adding an annotation to an OVO message that is forwarded to Service Desk. Optionally this rule takes care of the ownership of the OVO message. The rule is triggered when an incident is created where the creator's account name starts with OVOU and the source type is "OVO event". The rule executes the `ovsdupdatemesssage.sh` script through the Service Desk agent running on the host specified in the Host field of the account that created the incident.

- Database rule for incident `Send acknowledgement to OVO for UNIX`

  This database rule is used for acknowledging the OVO message when the corresponding incident is closed. The rule is triggered when an incident is modified where the creator's account name starts with OVOU and the source type is "OVO event" and the State is "Closed". The rule executes the `opcackmsg` command through the Service Desk agent running on the host specified in the Host field of the account that created the incident.

- Database rule for CI `OVO periodic outage from CI`

  This database rule is used to create and upload the outage file into OVO. The outage file contains information about when a node or service is in maintenance mode and therefore messages from this node or service must be suppressed by OVO. The rule is triggered when a configuration item has been created or changed. The planned outage information is retrieved from the schedules that are linked to the configuration item.

- Database rule for work order `OVO occasional outage from workorder`

  This database rule is used to create and upload the outage file into OVO. The outage file contains information about when a node or service is in maintenance mode and therefore messages from a node or service must be suppressed by OVO. The rule is triggered when a work order has been created or changed and the outage start and end are not empty. Messages from the configuration items that are linked to the work order will be suppressed during the occasional period.

- Database rule for schedule `OVO periodic outage from schedule`

  This database rule is used to create and upload the outage file into OVO. The outage file contains information about when a node or service is in maintenance mode and therefore messages from a node or service must be suppressed by OVO. The rule is triggered when a schedule has been created or changed. Messages from the configuration items that are linked to the schedule will be suppressed during the planned period.

- Database rule for schedule element `OVO periodic outage from schedule element`

  This database rule is used to create and upload the outage file into OVO. The outage file contains information about when a node or service is in maintenance mode and therefore messages from a node or service must be suppressed by OVO. The rule is triggered when a schedule element has been created or changed and the start and end are not empty and the schedule element is recurrent. Messages from the configuration items that are linked to the schedule containing the schedule elements will be suppressed during the planned period.

- Configuration exchange filters

  All filters with names starting with "OVO for UNIX" are used by configuration exchange to export all configuration data that are needed by the OVO for UNIX integration. The filters can be used to export data from one SD system to another SD system. For example, if customers have made changes to the OVO for UNIX configuration data in their test environment it is easy to export the data using the filters and subsequently import the data into their production environment.

- Configuration exchange filter group `OVO for UNIX`

  This filter group contains all the filters for the export of the OVO for UNIX integration configuration data. The filters are:

  — OVO for UNIX Roles

  — OVO for UNIX Accounts

  — OVO for UNIX Incident Source Type

  — OVO for UNIX Incident Template

  — OVO for UNIX Configuration Item Template

  — OVO for UNIX Mappings

  — OVO for UNIX DB Rules for Events

  — OVO for UNIX Filters

  — OVO for UNIX Filter Group

- OVO for UNIX Application
- OVO for UNIX Smart Action
- OVO for UNIX DB Rules for Occasional Outage from Work Order
- OVO for UNIX DB Rules for Periodic Outage from Schedule Element
- OVO for UNIX DB Rules for Periodic outage from Schedule
- OVO for UNIX DB Rules for Periodic Outage from CI

The order of the filters is important. The order of the filters determines the order of the import of the configuration data. For example, the OVO for UNIX role must be imported first before the OVO for UNIX account is imported because the account is connected to the role.

- Application `Opcmsg`

  This application is used to send a message to an OpenView Operations management server. The OVO agent must be installed and running on the Service Desk client computer. The `opcmsg` application calls the `opcmsg` command. The `opcmsg` command is part of the OVO agent.

  This application is used by the smart action `Send incident to OV Operations`.

- Smart action `Send incident to OV Operations`

  This smart action is used to send messages to an OpenView Operations management server. The smart action is defined for incidents. The incident fields Configuration Item, Severity and Description must be filled in for this smart action to work. The search code and name of the configuration item and the severity, description and information of the incident are supplied to the smart action that passes the information to the `opcmsg` application that in turn passes the information to the `opcmsg` command.

Configuration Data Uploaded into OVO

The configuration data is stored in the `ovsdintegrationovoconfig` directory and is uploaded automatically into OVO during the configuration phase using the `opccfgupld` utility of OVO.

The configuration data for running the OVO for UNIX Integration consists of the following:

- OVO User `Service Desk`

  This Operator will (optionally) own all messages that are forwarded to Service Desk through the Trouble Ticket interface.

  The SD-OVO integration uses the `ovsdovogetattributes` executable to get message attributes and the `ovsdupdatemessage` to add annotations to messages or to acknowledge messages. It performs this task by logging into OVO using the credentials of the Service Desk Operator. The credentials are stored in the `ovsdovo.conf` file. If you change the credentials of the Operator in OVO then you must also change the credentials in this configuration file accordingly.

  In order for the Operator to access all the necessary attributes, the Operator must have assigned responsibilities for all Node Groups and all Message groups.

  In order to add annotations, own or acknowledge messages the Operator must have all the Capabilities that are available.

- Message Group `ServiceDesk`

  All messages that are related to Service Desk processes belong to this Message Group.

- Application `Exchange Nodes` in the Application Group `Service Desk`

  This application calls the `ovsdexchovonodes.sh` script to export nodes from the Node Bank and import them as configuration items into OVO. The application runs as root on the OVO management server. It does not require a window for presentation.

- Application `Insert Incident` in the Application Group `Service Desk`

  This application calls the `ovsdevent.sh` script to forward one or more selected messages in the Message Browser to Service Desk. The additional parameter $OPC_MSGIDS_ACT is passed to the application. The application runs as root on the OVO management server. It does not require a window for presentation.

- Trouble Ticket System

The Trouble Ticket interface is enabled. The `ovsdevent.sh` script is called by the Trouble Ticket System. The system passes the message ID to the `ovsdevent.sh` script that in turn calls the `ovsdgetovoattributes` executable to retrieve more message attributes for creating an incident in Service Desk.

7 User Tasks

This chapter describes OVO integration tasks that need to be performed on a regular basis.

Exporting and Importing Services and Nodes from OVO into Service Desk

All nodes and services managed by OVO should have a corresponding configuration item in Service Desk. This task needs to be performed so that events coming from the OVO server can be linked with the correct configuration item in the Service Desk database.

NOTE The OVO management server is also a managed node.

Exporting and Importing Nodes and Services from OVO for Windows

Nodes can be imported from the OVO for Windows Management server as follows:

1. In the OpenView Console, navigate to the **Data** workspace in the **OV Configuration Workspace** group.
2. In the Data tree view, select **Unique Fields > Configuration Item** to open the Unique Fields dialog box, then clear the **Search code** check box.

   This ensures that nodes imported as configuration items are not rejected because they have non-unique search codes.
3. In the Data tree view, select **Data Exchange > Data Exchange Task**.
4. Select **File > New > Data Exchange Task**. The **New-Data Exchange Task** dialog box opens.
5. To export the nodes and services from OVO for Windows, select **Export**, navigate to the `<DataDir>\Datafiles\data_exchange\config` folder, using **Quick Find**, and open the `OvSdOvow.conf` file.
6. Specify the name and location of the Exchange file (temporary file) that will be populated with the results of the export process as defined in the respective configuration files.

7. Select **Import** and using **Quick Find**, select the **Application Account** that you want to use for importing (the account name should begin with `OVOW`).
8. Select the correct import mapping using **Quick Find**:
   - `OvoWindows` (import mapping for configuration items and services)
9. Save and close the **New-Data Exchange Task** dialog.
10. Right-click on the newly created Data Exchange Task and select **Start**.
11. The **Import Process** dialog box appears and shows the progress of the import/export process. When finished, close this dialog box.

For additional information about how to use Data Exchange for exporting and importing data, please refer to *HP OpenView Service Desk: Data Exchange Administrator's Guide*.

Importing Nodes from OVO for UNIX

The `ovsdexchovonodes.sh` script extracts the nodes from OVO and imports them into Service Desk. This script can be started from the application bank in the OVO user interface. The script calls `OvObsExporter` to export the node data into an XML file, and then calls `OvObsImporter` to import the node data from this XML file into Service Desk.

For additional information on modifying the import mapping, refer to *HP OpenView Service Desk: Data Exchange Administrator's Guide*.

Manually Creating a Configuration Item

You can manually create configuration items in Service Desk for all or part of your managed nodes. Refer to the Service Desk online help for detailed instructions on how to create a configuration item for a managed node:

1. Open a new **Configuration Item** dialog. This can be done by selecting **Configuration Item** from the shortcut bar or selecting New then **Configuration Item** from the **File** menu.
2. In the **Search code** field, enter the name of the managed node.
3. In the **Name** field, enter the Fully Qualified Domain Name (FQDN) for the managed node.

The Name field is used by the OVO Integration to identify the configuration item. The incident that is created by the OVO Integration relates the configuration item to the incident using the Name field. The configuration item corresponds to the node the incident originates from.

If the Name field does not contain the FQDN, the incident is still created, but the Configuration Item field of the incident will be empty, which makes the incident less useful.

Forwarding Messages

This section describes the message-forwarding functionality of the OVO Integration.

Manually Forward Messages from Service Desk to OVO

Smart actions are available for this integration. A smart action makes it possible for users to forward a message to the OVO management server to notify the OVO operators that a problem is detected.

To manually create an event in OVO based on an incident in Service Desk:

- In Service Desk, open an existing incident form, or create a new incident. Check that the severity field is not empty.
- On the menu bar, select **Action > Send Incident To OV Operations**.

The incident is then forwarded to OVO and is displayed in the OVO Message browser.

Manually Forwarding Messages from OVO to Service Desk (OVO for UNIX Only)

This feature makes it possible for users to manually forward a message to Service Desk. You may need to manually forward a message when:

- A message is created that was not detected by the automatic fault detection application
- A message is created that is not configured to be sent to Service Desk.

To manually forward messages:

1. Open the message browser in the OVO Console.
2. Select the messages you want to forward.
3. From the Application Bank select **Insert Incidents**. The selected messages are sent to Service Desk.

4. If during the configuration procedure you configured the SD operator to own the messages forwarded to Service Desk, the severity label turns white when a message is sent to Service Desk, showing that Service Desk now owns the message.

Outage Planning (OVO for UNIX Only)

You can prevent Service Desk from receiving OVO for UNIX server messages by letting OVO suppress the messages when an outage is planned. This is done by using mkoutage to download outage information from Service Desk and upload this information into OVO. The download and upload processes are performed automatically through the use of database rules in Service Desk. Two types of outages are recognized: periodic (or recurring outages) and occasional (or incidental outages). Outage planning is an administrative task in Service Desk, and it can only be used to suppress messages when installed with the OVO integration.

NOTE

It is not possible to automatically detect if outage periods manually entered on your OVO server application conflict with outage periods sent from Service Desk. If a conflict occurs, the result may be an error in the outage time period in the OVO application.

To define outages in Service Desk:

- To enter occasional planned outages in the work order dialog box, see the **Plan Incidental Outages** topic in the Online Help.
- To enter Periodic Outages in Service Desk, navigate to the **Configuration Item** form, select the **Downtime** tab, and click **New**. The **New-Schedule-CI** dialog appears. Select the previously defined schedule and select **Planned Downtime** from the **Type** drop-down list and click **OK**.

Viewing OVO Service State (OVO for Windows Only)

A browser can be used to view all services that are not working or just top level services that are not working, depending on the URL used. The services are color coded to show the level of importance: red equals critical; orange equals major; yellow equals minor; and light blue (Cyan) equals warning. The following scenarios are designed to give you an idea of how this feature might be used:

- A support manager wants an update on the state of the top level OVO services that are down. The manager clicks the Top OVO services shortcut on the desktop and gets a view of all top level services currently down.
- A specialist is working at a customer site and needs an update on the services that are down. The specialist enters the service viewer URL, in the browser of a computer with Internet access, and gets a view of all services that are down.
- The customer support center begins to receive numerous calls for service. To get a quick overview of services that are currently down the help desk engineer uses the Service viewer shortcut on the desktop to quickly locate the down service and answer the calls.

A OVO for Windows Files

This chapter lists the files installed with the OVO for Windows integration.

Installed Files

The following files are installed on the OVO management server (<InstallDir>=`C:\Program Files\HP OpenView`):

- `<InstallDir>\misc\sd\ovow\Monitor_LoadObject_error-logfile.policy`

  This file holds the actual policy for monitoring the log file that is created when an error occurs during alert sending to Service Desk. This policy will be uploaded to OVO for Windows.

- `<InstallDir>\misc\sd\ovow\Monitor_LoadObject_logfile.policy`

  This file holds the actual policy for monitoring the log file that is created during alert sending to Service Desk. This policy will be uploaded to OVO for Windows.

- `<InstallDir>\misc\sd\ovow\Monitor_Service_Desk_logfile.policy`

  This file holds the actual policy for monitoring the log file that is created by Service Desk. This policy will be uploaded to OVO for Windows.

- `<InstallDir>\misc\sd\ovow\Forward_message_changes_to_Service_Desk.policy`

  This file holds the actual policy that executes when a change is done on an OVO alert. This policy will then execute a rule that forwards the change of this alert to Service Desk and updates the representing incident. This policy will be uploaded to OVO for Windows.

- `<InstallDir>\misc\sd\ovow\Forward_messages_to_Service_Desk.policy`

  This file holds the actual policy that executes when an OVO alert is created. This policy will then execute a rule that creates an incident in Service Desk. This policy will be uploaded to OVO for Windows.

- `<InstallDir>\misc\sd\ovow\config.mm`

  This is the configuration file that holds the definition of all the policies that are to be used by OVO for Windows.

- `<InstallDir>\misc\sd\ovow\OvSdServiceExport.zip`

A command line tool that allows you to export information about an OVO for Windows service. Please see “Viewing OVO Services in a HTML Browser” on page 37.

- `<InstallDir>\lbin\sd\ovow\OvSdOvowControl.vbs`

  This is the script that is executed by the rule in the policies. This script generates the LoadObject command line to be executed so the alert is forwarded to Service Desk. This script converts the several OVO for Windows attributes to usable attributes for Service Desk.

- `<InstallDir>\lbin\sd\ovow\OvSdOvowCommandline.exe`

  Executes any string on the COMMAND environment variable. In this case it executes the command line generated by the LoadObject tool.

- `<InstallDir>\lbin\sd\ovow\OvSdOvowLoadPolicies.bat`

  This file uploads the policies to OVO for Windows. Deployment on the management server is needed after this.

- `<InstallDir>\lbin\sd\ovow\OvSdViewCi.bat`

  Shows you the configuration item that is given as a parameter to this file.

- `<InstallDir>\lbin\sd\ovow\ovsd_upgrade_to_510.bat`

  After the upgrade has been performed, this file needs to be executed. This file will rename the extensions of the old files to `*.old`. This enables you to modify any new files to preserve any configuration changes made to the previous installation.

Configuration Files

The configuration data for running the OVO for Windows integration consists of the following:

- `<DataDir>\Datafiles\data_exchange\xml\OvSdOvowConfig.xml`

  Configuration exchange file to be uploaded into Service Desk. This file contains configuration data needed by the OVO integration.

- `<DataDir>\Datafiles\data_exchange\config\OvSdOvow.conf`

  Configuration file for the data exchange export process. This file is used by the data exchange exporter utility. This file contains information how to connect to the OVO database. It also describes what data will be extracted from the OVO database. See Chapter 1 of *HP OpenView Service Desk Data Exchange Administrator's Guide* for a detailed description of this file.

- `<DataDir>\conf\obs\loadobject\OvObsLoadObject.conf`

  This file holds the configuration that is needed in order to insert information into Service Desk. Event forwarding information like server, account and data mapping that inserts the information is stored in this file.

Runtime Files

The following files are created during runtime of the integration (<DataDir>= `C:\Program Files\HP OpenView\Data`):

- `<DataDir>\Log\OvObsLoadObject_error.log`

  Log file that is created when an error occurs during forwarding of a message to Service Desk.

- `<DataDir>\Log\OvObsLoadObject.log`

  Log file that is created during forwarding of a message to Service Desk.

- `<DataDir>/log/obs/data_exchange/<importmappingname>_exp.log`

  The name of this log file is changeable. Log file containing messages generated during the export of the node information out of OVO.

- `<DataDir>/log/obs/data_exchange/<importmappingname>_imp.log`

  The name of this log file is changeable. Log file containing messages generated during the import of the node information into Service Desk.

- `<DataDir>/log/obs/data_exchange/<importmappingname>_imp_error.log`

  The name of this log file is changeable. Log file containing error messages generated during the import of the node information into Service Desk.

- `<DataDir>\Log\Systemx.0.en_US`

  Log file created by the OVO Management server where `x` can be a number from 1 to n.

B OVO for UNIX Files

This chapter lists the files installed with the OVO for UNIX integration.

Installed Files

The following files are installed on the OVO management server (<InstallDir>=/opt/OV):

- <InstallDir>/bin/ovsdovosetup.sh

  Configuration shell script that must be run after installation of the OVO integration. The script asks a few questions. The answers are used to modify configuration data before they are uploaded into Service Desk and OVO. The script creates configuration files in the <DataDir>/conf subdirectories based on the sample files located in the <InstallDir>/newconfig/conf/sd directory.

- <InstallDir>/lbin/sd/ovsdevent.pl

  Perl script used by ovsdevent.sh. The script formats the attributes of the OVO messages before they are uploaded into Service Desk.

- <InstallDir>/lbin/sd/ovsdevent.sh

  Shell script that is called by OVO's Trouble Ticket interface or by the "Insert Incident" application in the Application Bank. The parameters supplied to the script are one or more IDs of OVO messages. For every message the script retrieves the message attributes from OVO using the executable ovsdgetovoattributes. After formatting the attributes it calls the Object Loader to create an incident in Service Desk. The script needs ovsdevent.conf and ovsdovo.conf.

- <InstallDir>/lbin/sd/ovsdexchovonodes.sh

  Shell script that is called by the "Exchange Nodes" application in the Application Bank. The script exports node information of nodes in the Node Bank from the OVO database and imports the nodes as Configuration Items in Service Desk. The script uses the Exporter and Importer of Data Exchange. See the Data Exchange Administrator's Guide for a detailed description of these tools. The script needs ovsdexchovonodes.conf.

- <InstallDir>/lbin/sd/ovsdmkoutage.sh

Shell script that is called by the Service Desk agent triggered by database rules in Service Desk. The script retrieves outage data from Service Desk and creates an outage file that is uploaded into OVO. The script needs `ovsdmkoutage.conf`.

- `<InstallDir>/lbin/sd/ovsdupdatemessage.sh`

  Shell script that is called by the Service Desk agent triggered by database rules in Service Desk. The script calls the executable `ovsdupdatemessage` to update the OVO message corresponding with the incident in Service Desk. The update consists of adding an annotation to the message and optionally owning the message. This script needs `ovsdovo.conf`.

- `<InstallDir>/lbin/sd/ovsdgetovoattributes`

  Binary executable that is called by `ovsdevent.sh`. The executable connects to the OVO management server and retrieves attributes of a given message. The message ID is passed as parameter to the executable.

- `<InstallDir>/lbin/sd/ovsdupdatemessage`

  Binary executable that is called by `ovsdupdatemessage.sh`. The executable connects to the OVO management server and updates a given message. The message ID is passed as parameter to the executable.

- `<InstallDir>/lbin/sd/ovsdovoutils/*`

  Binary executables of `ovsdgetovoattributes` and `ovsdupdatemessage` for all supported platforms. The configuration script `ovsdovosetup.sh` determines which executables are needed and copies them to the `<InstallDir>/lbin/sd` directory.

- `<InstallDir>/newconfig/conf/sd/ovsdexchovonodes.conf`

  Configuration file that is used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/obs/data_exchange/ovsdexchovonodes.conf` file.

- `<InstallDir>/newconfig/conf/sd/ovsdevent.conf`

  Configuration file that is used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/obs/loadobject/ovsdevent.conf` file.

- `<InstallDir>/newconfig/conf/sd/ovsdovo.conf`

Configuration file that is used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/sd/ovsdovo.conf` file.

- `<InstallDir>/newconfig/conf/sd/ovsdmkoutage.conf`

  Configuration file that is used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/sd/ovsdmkoutage.conf` file.

- `<InstallDir>/newconfig/conf/sd/ovsdintegrationsdconfig.xml`

  Configuration Exchange file that is used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/obs/data_exchange/ovsdintegrationsdconfig.xml` file.

- `<InstallDir>/newconfig/conf/sd/ovsdintegrationovoconfig/*`

  Configuration data files that are used by the `ovsdovosetup.sh` configuration script to create the `<DataDir>/conf/sd/ovsdintegrationovoconfig/*` files.

- `<InstallDir>/java/obsmodules/sd-outage.jar`

  Java Archive file that contains Java classes used by the `ovsdmkoutage.sh` script.

Configuration Files

The following files are created by the Configuration Script ovsdovosetup.sh (<DataDir>=/var/opt/OV):

- <DataDir>/log/ovsdovosetup.log

  Log file containing information and error messages from the ovsdovosetup.sh script.

- <DataDir>/conf/obs/data_exchange/ovsdexchovonodes.conf

  Configuration file for ovsdexchovonodes.sh. This file is used by the Data Exchange Exporter utility. This file contains information how to connect to the OVO database. It also describes what data will be extracted from the OVO database. See Chapter 1 of *HP OpenView Service Desk Data Exchange Administrator's Guide* for a detailed description of this file.

- <DataDir>/conf/obs/loadobject/ovsdevent.conf

  Configuration file for ovsdevent.sh. This file contains information on how to connect to the Service Desk management server. It also describes which import mapping to use for inserting an incident into Service Desk. See Chapter 2 of *HP OpenView Service Desk Data Exchange Administrator's Guide* for a detailed description of this file.

- <DataDir>/conf/sd/ovsdovo.conf

  Configuration file for ovsdevent.sh and ovsdupdatemessage.sh. This file contains information how to connect to the OVO management server. It also describes if messages should be owned by the SD operator in OVO after forwarding the message to Service Desk.

- <DataDir>/conf/sd/ovsdmkoutage.conf

  Configuration file for ovsdmkoutage.sh. This file contains information how to connect to the Service Desk management server. It also describes how to check the syntax of the outage file and how to upload it into OVO.

- <DataDir>/conf/obs/data_exchange/ovsdintegrationsdconfig.xml

Configuration Exchange file to be uploaded into Service Desk. This file contains configuration data needed by the OVO integration. The upload is either performed automatically by the `ovsdovosetup.sh` configuration script or must be uploaded manually using a Service Desk client.

- `<DataDir>/conf/sd/ovsdintegrationovoconfig/*`

  Configuration data files to be uploaded into OVO using the `opccfgupld` utility. This directory contains files that are created by the `opccfgdwn` utility. The files contain configuration data needed by the OVO integration. The upload is performed automatically by the `ovsdovosetup.sh` configuration script.

Runtime Files

The following files are created during runtime of the integration (<DataDir>=/var/opt/OV):

- <DataDir>/tmp/ovsdevent_<time stamp>.tmp[1-2]

  Temporary files that are created by ovsdevent.sh and removed when no longer needed.

- <DataDir>/log/ovsdintegrationsdconfig.log

  Log file containing messages of the automatic upload of the ovsdintegrationsdconfig.xml file into Service Desk by the ovsdovosetup.sh configuration script.

- <DataDir>/log/ovsdexchovonodes.log

  Log file containing messages produced by the ovsdexchovonodes.sh script.

- <DataDir>/log/obs/data_exchange/ovsdexchovonodes_<time stamp>_exp.log

  Log file containing messages generated during the export of the node information out of OVO.

- <DataDir>/log/obs/data_exchange/ovsdexchovonodes_<time stamp>_imp.log

  Log file containing messages generated during the import of the node information into Service Desk.

- <DataDir>/log/ovsdevent.log

  Log file containing information messages generated by the ovsdevent.sh script.

- <DataDir>/log/ovsdevent_error.log

  Log file containing error messages generated by the ovsdevent.sh script.

- <DataDir>/datafiles/data_exchange/ovsdexchovonodes_<time stamp>.xml

Intermediate XML file produced by the data exchange exporter utility. The file contains node information from the OVO database. The file is read by the data exchange importer utility to import the node information into Service Desk.