HP OpenView Smart Plug-in for Databases

For HP OpenView Operations for UNIX

Software Version: 10.40

Configuration Guide

Document Release Date: November 2006
Software Release Date: November 2006

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

Trademark Notices

Microsoft® and Windows NT® are US registered trademarks of Microsoft Corporation.

Pentium® is a US registered trademark of Intel Corporation.

UNIX® is a registered trademark of The Open Group.

Oracle® is a registered trademark of Oracle Corporation.

Documentation Updates

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

- Software 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 of a document, go to:

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.

Support

You can visit the HP OpenView Support web site at:

www.hp.com/managementsoftware/support

HP OpenView online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into 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 sign in. Many also require a support contract.

To find more information about access levels, go to:

www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

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

Contents

1 An Overview of the Database SPI

This chapter covers basic concepts necessary to understanding the Database Smart Plug-Ins (DB-SPI). It includes the following:

- Introducing the Smart Plug-in for Databases
- How the Smart Plug-in for Databases Works
- Components of the Smart Plug-in for Databases

Introducing the Smart Plug-in for Databases

The Smart Plug-in for Databases (DB-SPI) adds Oracle, Sybase, Informix, and Microsoft SQL Server database monitoring capabilities to HP OpenView Operations (OVO).

Smart Plug-In tools: Used in conjunction with OVO, DB-SPI offers centralized tools that help you monitor and manage database instances. From the OVO console, an operator can apply the same familiar HP OpenView performance and problem managing processes to databases. DB-SPI metrics can be graphed using OpenView Performance Manager or PerfView, either of which can help you to analyze trends in database use and performance across your organization.

Smart Plug-In data: After completing the DB-SPI installation and configuration, you can find hundreds of key database metrics and event monitors. These metrics and event monitors for Oracle, Sybase, Informix and MS SQL Server cover the following areas:

- Database status
- Space management
- Performance
- Archive/Trace
- Transaction logging
- Security
- Errors

Smart Plug-In uses/customizations: Database administrators can choose those metrics or event monitors that are most crucial to the enterprise by modifying DB-SPI templates (within a local file stored on the managed node or from the OVO console) to precisely meet their needs. Within the OVO templates, incoming data is measured against predefined rules that generate useful information in the form of messages and severity level alerts. These messages and alerts can be reviewed for problem analysis and resolution. Corrective actions that are pre-defined for specific events or threshold violations can be automatically triggered or operator-initiated.

How the Smart Plug-in for Databases Works

DB-SPI messaging, reporting, and action-executing capabilities are based on the OVO concept of template monitors. These template monitors define conditions within the database that help you avoid potential problems or resolve those that occur. As a result, you can avoid serious disruptions to database use.

How DB-SPI Collects and Interprets Database Performance/Availability Information

The DB-SPI templates are grouped so that you can find them easily. The *monitor templates* define metric conditions and the *collector templates* list targeted metric values to collect. Each *collector template group* is assigned to a collection interval of **5 minutes, 15 minutes**, **one hour**, or **one day.** The collection interval indicates how often data is collected for that specific group.

Controlling data collection for the group is the job of the collector template. Specifically, the collector template has two functions: (1) to run the collector/analyzer at each collection interval and (2) to list all monitors (predefined metric conditions) within the group. Each monitor template determines how the actual metric condition is handled according to the definition of the monitored object, the threshold for that monitored object, the message text that is triggered when the threshold is exceeded, the actions to execute, and the instructions that appear.

How DB-SPI Displays the Information

Topology and service alerts in the Service Navigator: The Database SPI integrates with Service Navigator, showing installed database instances and any availability/performance issues. Service problems are apparent when an instance on the map turns red.

Messages in the Message Browser: Using the monitor template settings and the database monitor values that the collector/analyzer has gathered, OVO agent software forwards the appropriate messages to the OVO console, where they are displayed with severity level color-coded in the Message Browser.

Instruction Text: Messages generated by DB-SPI programs contain instruction text to help diagnose and remedy problems. Corrective actions that are pre-assigned to events can be triggered automatically or manually by an operator.

Reports: Some reports are generated automatically, and others can be generated by operator action using the Message Browser or using the OVO Application Bank.

Graphs: Metrics collected by DB-SPI can be graphed using the (new) OpenView Performance Manager or PerfView. Database SPI metric values can then be viewed for trend analysis.

Changes you can make

You can use DB-SPI templates with no customization, or you can change them as you find necessary. Minor modifications and major customizations are listed here:

Modification of Default Templates: You can change a default template by using the OVO console to select the template and change conditions within the template. You can change: (1) collection interval, (2) threshold, (3) message text, (4) duration, (5) severity, (6) actions.

Threshold values can be modified at the OVO console for global changes, or they can be defined within a file that is stored on the managed node. This file overrides the template threshold values. Please see Overriding OVO Thresholds by Configuring Local System Thresholds on page 76 for information and instructions on configuring local threshold values.

Creation of Custom Template Groups: You can create custom template groups from the default templates. The simplest method is to copy the default template group, modify each monitor and collector template as needed, rename each using the *tag parameter* in the collector template, and assign them. Complete instructions for creating such templates are located in Creating Custom Templates for Large Scale Environments on page 59.

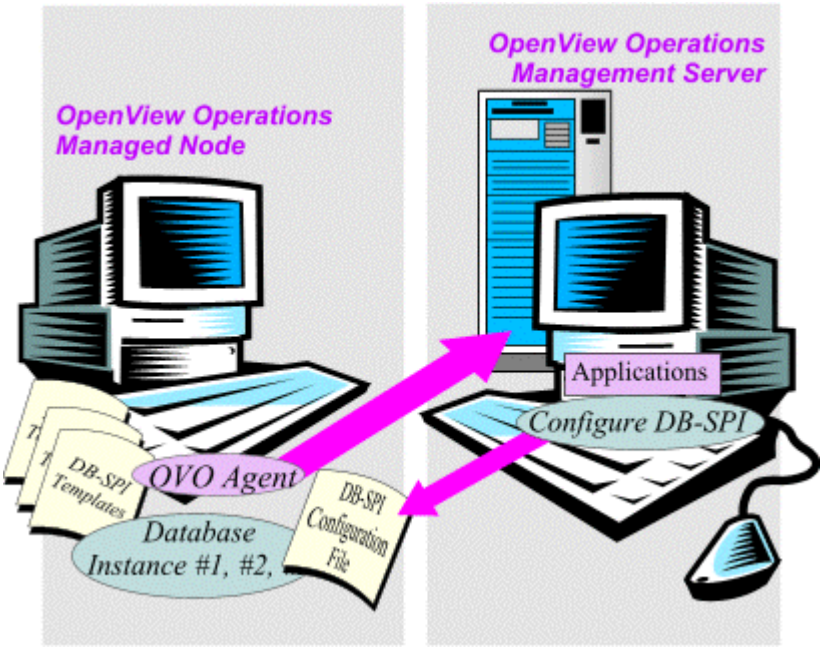
Some *filters* can also be set in DB-SPI by appending WHERE clause fragments to SQL queries sent by the DB-SPI collector/analyzer to system tables. For more information on this and other customization topics, see Chapter 3, Using and Customizing the Database SPI.

Custom Metrics: The ability to define your own metrics or user-defined metrics (UDMs) is a powerful feature that you can use to expand the monitoring capabilities of DB-SPI. Complete instructions for Oracle, MS SQL Server, Informix, and Sybase user-defined metrics are available in Chapter 6, User Defined Metrics.

Components of the Smart Plug-in for Databases

The Database Smart Plug-In components include applications and templates that allow you to configure and receive data in the form of messages and short reports. These messages (available in the OVO Message Browser) and short reports (through DB-SPI applications or message details) provide you with information about conditions present in database instances on specific managed nodes.

Figure 1 DB-SPI components on the managed node and the OVO Management Server



The above diagram shows the OVO agent transferring data to the OVO management server. It also shows the DB-SPI configuration file and templates. The configuration file allows DB-SPI to make the connection to the database on the managed node, while the DB-SPI templates direct what database data to gather and how to interpret it.

DB-SPI configuration applications let you connect to named database instances on specific managed nodes. After you have configured database instances and managed nodes, you can assign templates to the nodes. With OVO agent software running on the node(s), you can use DB-SPI reporting applications to generate reports. In addition, if you use OpenView Performance Manager or PerfView, you can generate graphs that show DB-SPI data.

Chapters 2 instructs you on how to install the components and use them to configure database instances, managed nodes, and the data collections (templates) which you want to use. The sections that follow here and Appendix B give you brief descriptions of all components.

Applications

DB-SPI applications include configuration, report, and service discovery generating utilities. In the Application Bank window, all Database SPI applications can be accessed through the top level group: **DB-SPI**. Those applications are as follows:

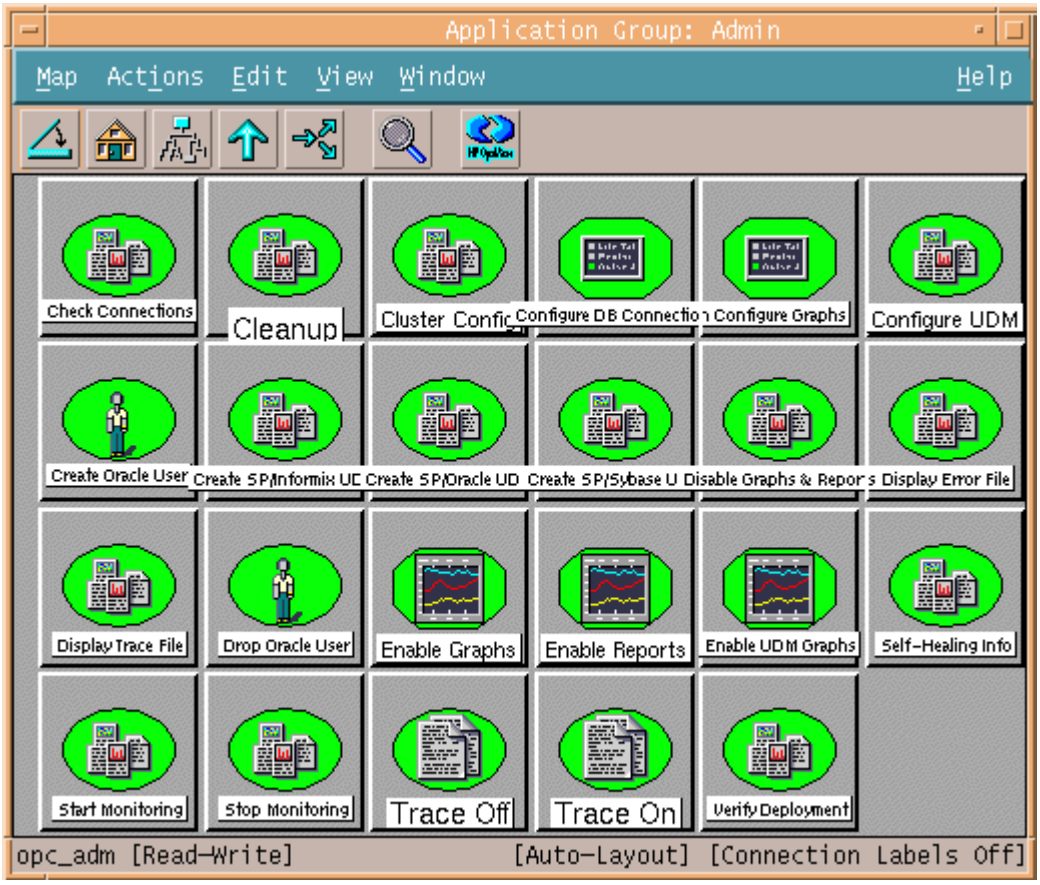
Service Discovery This group includes a service discovery application for each database type that can assist you in configuring the Database SPI connection to every database instance you want to monitor. In addition, Service Discovery also populates the Service Navigator map (available if you have installed Service Navigator), showing discovered instances and their color-coded availability/performance status.

Admin and Admin Windows The *Admin* and *Admin Windows* groups contain identical applications, with the Admin group appropriate for UNIX managed nodes and *Admin Windows* appropriate for Windows nodes. These applications allow you to configure the connection to the database you are monitoring on the managed node. Because the Admin group contains configuration and troubleshooting applications requiring "root" permission, it is recommended the group be assigned to the OVO administrator. This group contains the following applications:

- **Check Connections** Checks the connection to configured databases.
- **Cleanup** Removes runtime files like trace, log, and persistent store files.
- **Cluster Config** Prepares APM XML files and places them in appropriate folder.
- **Configure DB Connections** Opens file with suggested syntax for configuring the managed node database. File entries define the database location, database account used by the DB-SPI to access the database; can also contain other settings for files/programs, such as the trace file and filtering parameters (also available for Windows managed nodes).
- **Configure UDM** Opens the user-defined metrics configuration file with suggested syntax for configuring UDMs.
- **Create SP/MSSQL UDM** Used to create stored procedures for MS SQL Server.
- **Create Oracle User** (optional) Used to define Oracle user/password to access specific database instances on the managed node (must be implemented prior to DBSPI-Config, where database instances are defined).
- **Create SP Informix UDM** Opens a file, where you can configure stored procedures for the creation of Informix user-defined metrics.
- **Create SP Oracle UDM** Opens a file, where you can configure stored procedures for the creation of Oracle user-defined metrics.
- **Create Sybase SP UDM** Opens a file, where you can configure stored procedures for the creation of Sybase user-defined metrics.
- **Disable Graphs & Reports** Disables performance data collections including those used for graphing, UDM, and Reporter.
- **Display Error File** Displays the contents of the DB-SPI error file:
  `/var/opt/OV/dbspi/log/dbspierror`
  *Paths for specific operating systems are shown in Appendix A.
- **Display Trace File** Displays the last 1,000 lines (configurable) of the trace file.
- **Drop Oracle User** Removes the current Oracle logon user/password definition for a specific database instance.
- **Enable Graphs.** Enables metric data collection for performance graphs.
- **Enable Reports** Enables metric data collection for Reporter (Reporter templates must be deployed to complete the implementation).
- **Enable UDM Graphs** Enables metric collections for UDM graphing purposes (using the separately purchased OpenView Performance Manager).

- **Configure Graphs** Opens the configuration file for the separately purchased OpenView Performance Manger, in which you can insert an entry for the location of your Web browser.
- **Migrate Agent** When run on the managed node, prepares for changing the agent, either from the DCE agent to the new HTTPS agent, or from HTTPS back to DCE. Run the application on the managed node before deploying the different agent.
- **Self-Healing Info** When run on the managed node, collects error message-related and OpenView Operations information that can be sent to HP support to troubleshoot SPI issues. This information is in the form of command output, showing configuration, log, and trace file contents. Please see Chapter 8, Error Messages & Troubleshooting for its use.
  Note: Your HP support contract entitles you to access cross-referenced information through the *HP Self-Healing Services* web site, linking you to a database where similar information and subsequent problem solutions increase troubleshooting effectiveness. Please see the *SPI DVD Installation Guide* and the section covering Self-Healing Services for more details.
- **Set Path** Adds actions, monitor, and commands directories to the PATH variable for DOS command prompt execution of DB-SPI programs. This application is only for the Microsoft Windows NT operating system.
- **Start Monitoring** Re-enables performance metric data collection and alert notification. To turn collection back on at a command line:
  UNIX: **/<*OVO_commands_directory*>*/dbspicol -v ON**
  Windows: **\<*OVO_commands_directory*>*\dbspicol -v ON**
- **Stop Monitoring** Temporarily disables performance metric data collection and alert notification. This application can be used to disable the metric collection during database maintenance, when using HP Serviceguard, or to minimize errors in the Message Browser if a database goes down unexpectedly. To turn collection off at a command line:
  UNIX: **/<*OVO_commands_directory*>*/dbspicol -v OFF**
  Windows: **\<*OVO_commands_directory*>*\dbspicol -v OFF**
- **Trace Off** Turns DB-SPI tracing off.
- **Trace On** Turns DB-SPI tracing on.
- **View Graphs** Launches your Web browser (that you configured during the installation) so that you can select and view a graph, using (separately purchased) OpenView Performance Manager.
- **Verify Deployment** Shows deployed files, versions, number of templates, defaults file, and also performs a connection check.

Figure 2 To open the Database SPI administrative task applications, from the Window menu select Applications Bank→DBSPI→Admin. These applications let you configure database connections, create user-defined metrics, generate graphs, start and stop collections, and troubleshoot

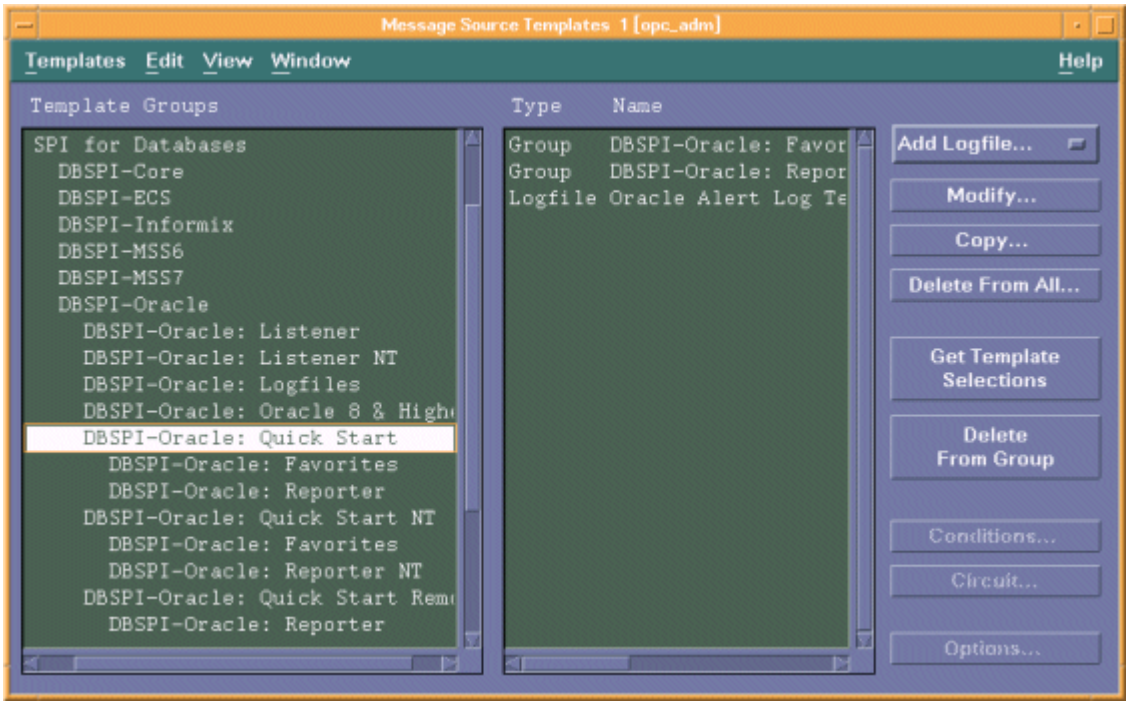


Templates

DB-SPI templates, in the *SPI for Databases* template group, are as follows:

- **DBSPI-Core** This group contains the agent data feed, message file monitor, and file system monitor for DB-SPI.
- **DBSPI-Informix** This group contains the message source templates for Informix.
- **DBSPI-Oracle** This group contains the message source templates for Oracle.
- **DBSPI-Sybase** This group contains the message source templates for Sybase.
- **DBSPI-MSS6 & DBSPI-MSS7** This group contains the message source templates for MS SQL Server.

Figure 3 DBSPI templates in the Message Source Template Window



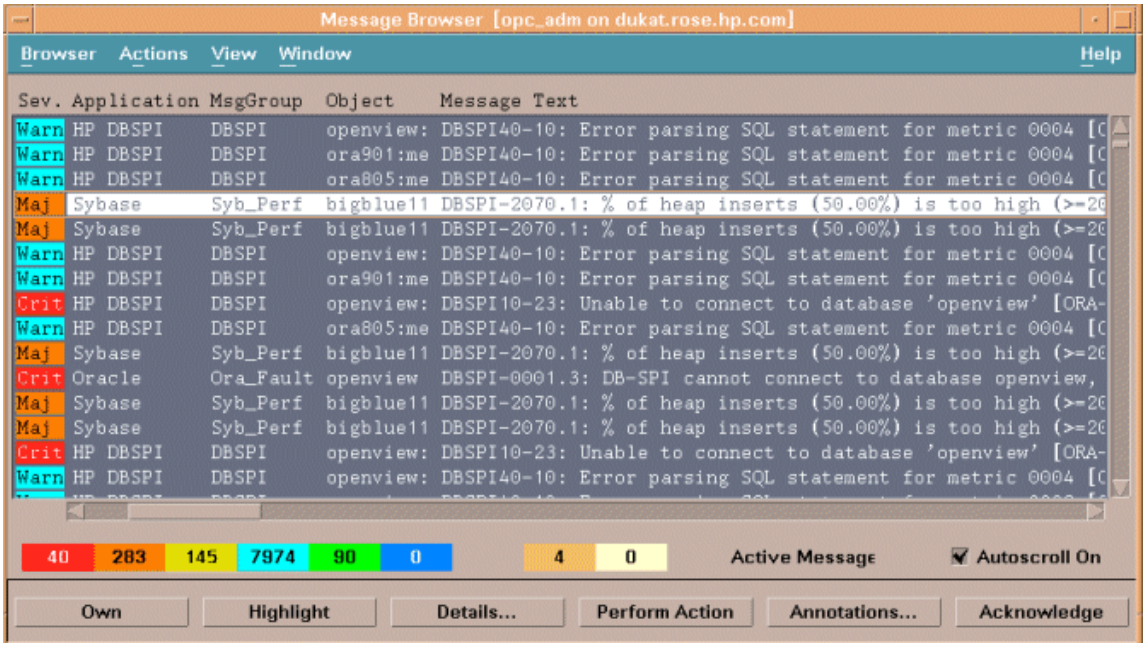
Message Groups

While DB-SPI templates are used to generate messages, DB-SPI Message Groups are used to classify the generated message. DBSPI messages can alert you to performance problems or error conditions. You may choose to assign one message type to one OVO operator and another type to an OVO administrator so that only the assigned message type appears in the Message Browser.

In order to indicate the message type, each message is assigned to a message group. Each DB-SPI message that appears in the Message Browser shows the message group to which it is assigned in the column adjacent to it. You can assign messages to the following groups:

Administrative Messages	Ora_Admin Inf_Admin Syb_Admin MS SQL_Admin
Configuration Messages	Ora_Conf Inf_Conf Syb_Conf MS SQL_Conf
Performance Messages	Ora_Perf Inf_Perf Syb_Perf MS SQL_Perf

Figure 4 The highlighted message here belongs to the Message Group Syb_Perf, indicating a performance issue, involving an alarm condition rated as Major



Reports

DB-SPI reports can show information on past or current conditions in the database. These reports can be triggered by alarm conditions or generated when you need information on the current state of the database.

Reports generated through *Automatic* or *Operator* actions are context sensitive, relating only to a single database instance on the managed node. Reports generated through the *Application Bank* show information on all database instances on a managed node.

Specifically, each report type is defined as follows:

- **Automatic Action Reports:** generated automatically by an alarm condition in relation to an *individual database instance* on a managed node. These reports are available by clicking the **Annotations** button in the OVO Message Browser. In the Message Browser the existence of this report is indicated by an *S* under the A column in the SUIAONE column. The *S* indicates that the report was generated successfully and is waiting in the Annotations of the message.
- **Application Bank Reports:** generated manually by dragging the managed node onto the report in the Application Bank. These reports reflect the current condition of *all configured database instances of the specific database type* on the managed node at the time you generate it.

  DB-SPI Oracle Application Bank Report: You can generate a report on all database instances on a managed node by dragging the node onto a report in the Application Bank window

```
Report For Database ora901
                      Tue Dec 11 09:52:48 2001

                   Metric TblSpCantExtCnt (0003)

                     FREE SPACE      FILE SYSTEM
```

```
TABLESPACE_NAME       EXTENT COUNT    EXTENT COUNT  AUTOEXTEND
--------------------- -------------  ------------  ----------
CWMLITE                 224             176615        YES
DRSYS                   196             176615        YES
EXAMPLE                   3              11038        YES
SYSTEM                   29              10092        YES
UNDOTBS                4138             176615        YES
TABLESPACE_NAME:     Name of the tablespace
EXTENT THRESHOLD:    Number of extents that must be available
FREE SPACE
 EXTENT COUNT:       Total number of extents available in
                     dba_free_space
FILE SYSTEM
 EXTENT COUNT:        Total number of extents available in the
                     file system (if AUTOEXTEND is enabled)
AUTOEXTEND:          Whether the tablespace has at least one
                     datafile where AUTOEXTEND is enabled.
```

Graphing Data

If you use OpenView Performance Manager or PerfView, you can generate graphs from most DB-SPI alarm messages by pressing the Perform Action button from the message or from the OVO Message Browser. This displays a graph of the metric that generated the message as well as other metrics that may be related to it.

In addition, data from any DB-SPI database instance can be graphed using OpenView Performance Manager or PerfView and data from DB-SPI. For more information on graphing with PerfView, see Chapter 4.

2 Installing and Configuring the Database SPI

This chapter covers installing, configuring, and removing DB-SPI program components for use with HP OpenView Operations (OVO). See Appendix B for a list of the DB-SPI components added during the installation process.

This chapter covers:

- Installation, Discovery, and Configuration
- Deinstallation

Installation & Configuration

Support Information: Details on supported operating systems are available in the Databases SPI Release Notes, located on the DVD you received under `/OV_DOC/DB_SPI<version>/release_notes.pdf`.

Agent configurations

Older versus newer versions of OVO agents: The Smart Plug-in for Databases (DB-SPI) continues to work with earlier versions of OpenView Performance Agent (also known as MeasureWare Agent), which it can detect during the installation. As a result, your new DB-SPI installation fully supports the graphing capabilities of PerfView.

Overriding newer versions with older versions: You can re-configure DB-SPI to work with the HP OpenView Performance subagent, included with OpenView Operations. This configuration does not support PerfView. To override the default configuration, which uses OpenView Performance Agent (MeasureWare Agent) when present, please see Changing from OpenView Performance Agent to the OpenView Subagent on page 75.

Switching DCE or HTTPS 8.0 agents to HTTPS or DCE: If for any reason you want to change from *HTTPS agent 8.1* to *DCE agent 8.1* or vice versa, refer to the *Smart Plug-in for Databases Release Notes* for the required application you run on each managed node for a successful changeover.

Required Tasks

If this installation is an upgrade of a previous installation, complete the steps in the sections Starting cleanup on managed nodes on page 39 and Removing software from the management server on page 39 to remove DB-SPI. After completing the steps to remove DB-SPI, proceed with the tasks described in this section.

- Install on the Management Server on page 22

- Add nodes to the appropriate DB-SPI node group on page 24
- Deploy Actions/Monitors/Commands to DB-SPI Node Group(s) on page 24
- Assign/Deploy the Discovery template(s) to the management server. on page 26
- Update User Responsibilities on page 27
- Specify operators for Service Map viewing on page 28
- Run Discovery on page 29
- Configure the database connection and (if appropriate) enable graphing/reporting on page 31
- Added configuration for OVO 8.x DCE Nodes on page 34
- Run the Service Discovery application again (see Task 6) on page 37
- Distribute DB-SPI templates on page 38

Task 1: Install on the Management Server

Install the Database SPI, using the *HP OpenView Smart Plug-ins DVD*. Refer to the *Smart Plug-ins DVD Installation Guide* for products/documentation locations on the DVD.

1 To install from the OVO Smart Plug-ins DVD:

For HP-UX systems, follow the instructions in the *Smart Plug-ins DVD Installation Guide* for mounting the DVD;
For Solaris systems, the DVD is automatically mounted.

2 Change to the directory according to management server installed OVO version and operating system as follows:

Table 1 Supported OVO Version

OVO version and platform	DVD	DIRECTORY
OVO 7 or OVO 8.1 on HP-UX 11.x	HP-UX DVD	/OV_DEPOT/
OVO 7 or OVO8.1 on Solaris	Solaris DVD	/OV_DEPOT/

3 To install all filesets, run `swinstall`; for example:

```
/usr/sbin/swinstall -s /OV_DEPOT/11.0HPUX.sdtape DBSPI (for HP-UX)
/usr/sbin/swinstall -s /OV_DEPOT/SOLARIS.sdtape DBSPI (for Solaris)
```

OR

To install a selected DB-SPI bundle (as listed in the following table) run `swinstall` with the name included. For example (type as one continuous line):

For HP-UX:
```
/usr/sbin/swinstall -s /OV_DEPOT/11.0HPUX.sdtape DBSPIOracleHPUX
```

For Solaris:
```
/usr/sbin/swinstall -s /OV_DEPOT/SOLARIS.sdtape DBSPIOracleSOL
```

Table 2 DB-SPI Software Bundles

Name	Version	Information
DBSPIInfAIX	10.40	HP OpenView DB-SPI: Informix (AIX only)
DBSPIInfAll	10.40	HP OpenView DB-SPI: Informix (All Platforms)
DBSPIInfHPUX	10.40	HP OpenView DB-SPI: Informix (HP-UX only)
DBSPIInfSOL	10.40	HP OpenView DB-SPI: Informix (Solaris only)
DBSPIMSSQL	10.40	HP OpenView DB-SPI: MSSQL Server
DBSPIOracleAIX	10.40	HP OpenView DB-SPI: Oracle (AIX only)
DBSPIOracleAll	10.40	HP OpenView DB-SPI: Oracle (All Platforms)
DBSPIOracleHPUX	10.40	HP OpenView DB-SPI: Oracle (HP-UX only)
DBSPIOracleLINUX	10.40	HP OpenView DB-SPI: Oracle (Linux only)
DBSPIOracleNT	10.40	HP OpenView DB-SPI: Oracle (NT only)
DBSPIOracleOSF	10.40	HP OpenView DB-SPI: Oracle (Tru64 only)
DBSPIOracleSOL	10.40	HP OpenView DB-SPI: Oracle (Solaris only)
DBSPISybaseAIX	10.40	HP OpenView DB-SPI: Sybase (AIX only)
DBSPISybaseAll	10.40	HP OpenView DB-SPI: Sybase (All Platforms)
DBSPISybaseHPUX	10.40	HP OpenView DB-SPI: Sybase (HP-UX only)
DBSPISybaseLINUX	10.40	HP OpenView DB-SPI: Sybase (Linux only)
DBSPISybaseSOL	10.40	HP OpenView DB-SPI: Sybase (Solaris only)

For Cluster and HP Serviceg configurations: If you run the OVO Management Server as a package within a cluster environment, repeat Tasks 1 and 2 on all cluster nodes that are capable of running the package. The OVO Management Server package has to be running on the cluster node while you run `swinstall`. You will receive warnings about already existing template groups, templates, conditions, which you can ignore. Warnings are listed in the `swagent.log` file.

Task 2: Add nodes to the appropriate DB-SPI node group

In this procedure you add nodes to the appropriate node group so that policy deployment can be done in one step, rather than node by node. Each node group has the database-type Quick Start template group already assigned to it.

1 At the console, start OpenView Operations (type `opc`) and log on as the OVO administrator.

2 Using the Window menu, open both the **Node Bank** and the **Node Group Bank** windows

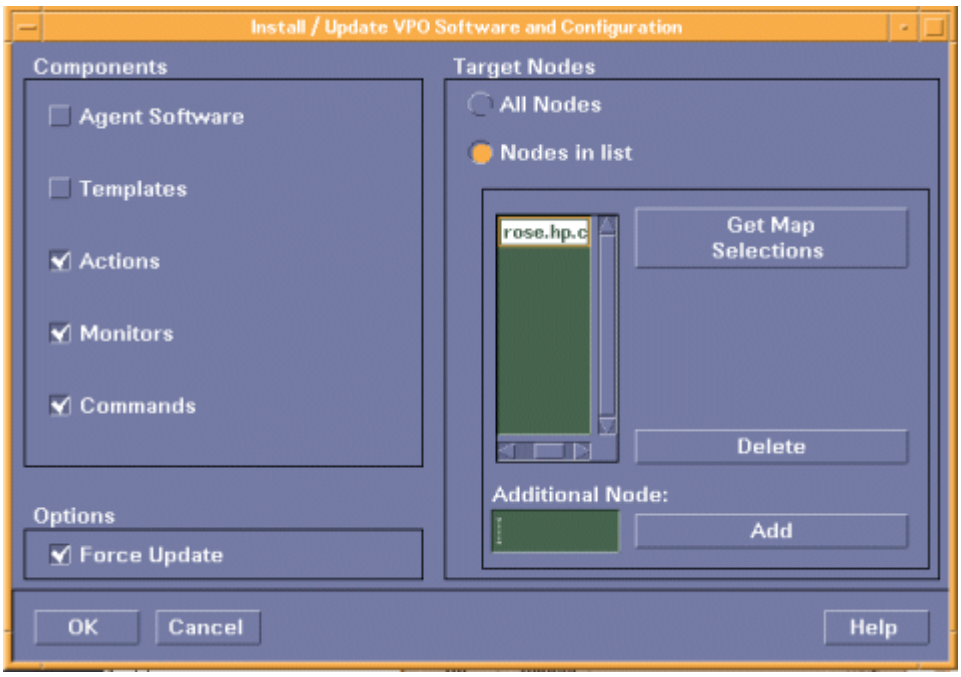
3 As appropriate, add each node to the proper node group by dragging and dropping from the Node Bank window to the Node Group Bank window:
- Informix
- Sybase
- MSSQL
- Oracle (NT)
- Oracle (Unix)

Task 3: Deploy Actions/Monitors/Commands to DB-SPI Node Group(s)

In the Node Group Bank window select the node groups, from among Informix, Sybase, MSSQL, Oracle (NT), and Oracle (Unix), to which you just added nodes and from the Actions menu select **Agents→Install/Update SW & Config**.

4 In the Install/Update OVO Software and Configuration window select check boxes for these components (do NOT select Templates):
- Actions
- Monitors
- Commands

Figure 5 The Install/Update Software and Configuration Window



5 Click the **Nodes in List** option button.

6 Click the **Force Update** check box.

7 Click **OK** to distribute Actions, Commands and Monitors to the managed node(s). The following message appears in the Message Browser:

The following configuration information was successfully distributed:
Actions Commands Monitors <node name(s)>

Task 4: (as desired for UNIX nodes):

Part A — Configure the OVO agent to run as non-root

To run the Database SPI under a non-root agent user, manual configuration steps are necessary. In those steps a file is generated that contains a list of users and commands accessed by DB-SPI. After the file is generated, you can edit it so that when necessary you can still run the Database SPI applications.

1 At the managed node, log on as `root`.

2 Set path by entering:
`. <OVO_commands_directory>/dbspisetpath`

(note blank space follows period [.])

3 Enter the following to run the script that generates the file (`dbspi_su`) that you will edit:
`dbspi_perl <OVO_commands_directory>/dbspi_root.pl`

See Running DB-SPI Applications from the Command Line on page 60 for the value of *<OVO_commands_directory>*.

Part B — Edit File to Enable use of Database SPI Applications.

The file in the following procedure can be modified only by a superuser.

1 At the managed node log on as `root` and open the file: `/etc/dbspi.su`

2 Edit the file by uncommenting lines and adding:
<user>:*<commands>*
as in the following examples:
`oracle:opt/oracle/product/sqlplus /nolog`
(allows **sqlplus** commands)
or
`oracle:opt/oracle/product/*`
(allows execution of all commands by Oracle user)

Task 5: Assign/Deploy the Discovery template(s) to the management server.

In this task you assign the discovery template relevant to the database type you are monitoring. This schedule template will then run discovery every night and update the Service Navigator map.

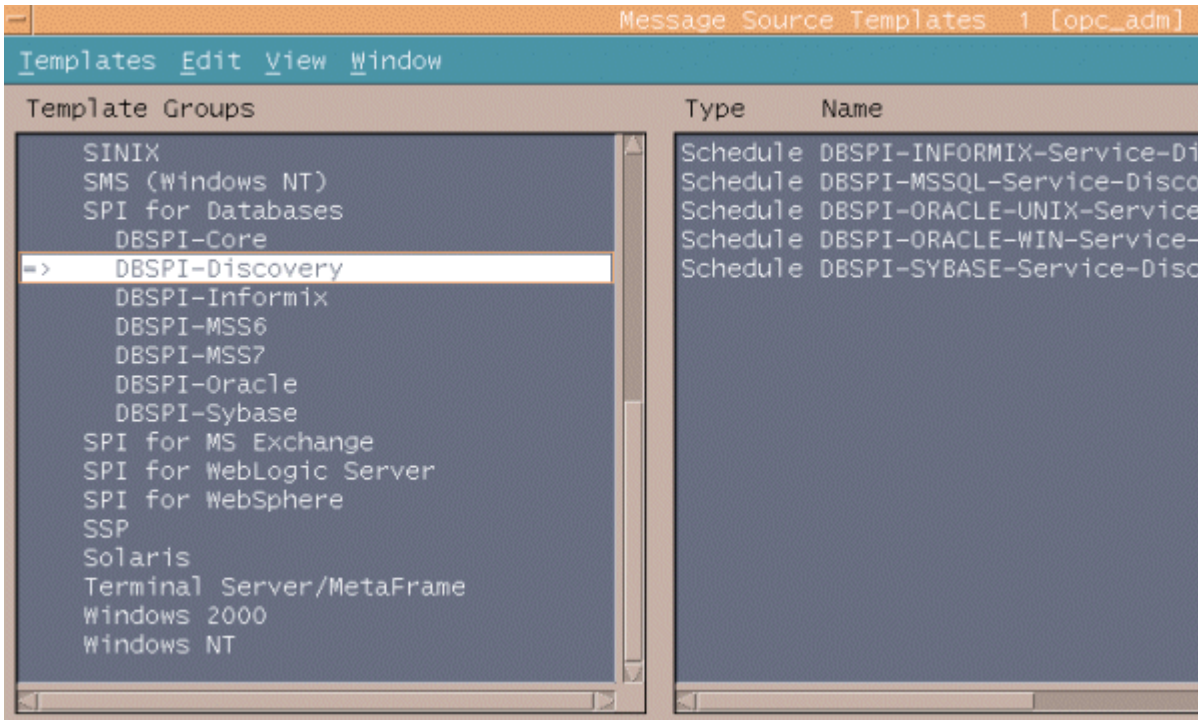
1 Open the Node Bank window and select the <*OVO_management_server*>; then from the Actions menu select **Actions→Agents→ Assign Templates...**.

2 In the Define Configuration window Node/Group column click the **Add...** button.

3 In the Add Configuration window click the **Open Template Window** button.

4 In the Message Source Templates window select **SPI for Databases→DBSPI-Discovery** and choose the appropriate service discovery template from:
DBSPI-Informix-Service-Discovery-Daily,
DBSPI-MSSQL-Service-Discovery-Daily,
DBSPI-Oracle-UNIX-Service-Discovery-Daily,
DBSPI-Oracle-WIN-Service-Discovery-Daily, or
DBSPI-Sybase-Service-Discovery-Daily

Figure 6 Service Discovery templates, according to database type are contained in the DBSPI-Discovery subgroup



5 Switch back to the Add Configuration window and click the **Get Template Selections** button; then click **OK**.

6 To distribute the (already) assigned template, from the Actions menu select **Agents→Install/Update SW & Config**.

The following message is displayed in the Message Browser:

```
The following configuration information was successfully distributed:
Templates
```

Task 6: Update User Responsibilities

To receive database-related messages from DB-SPI managed nodes, assign *DBSPI* and specific database type message groups as explained. Repeat the procedure for each operator.

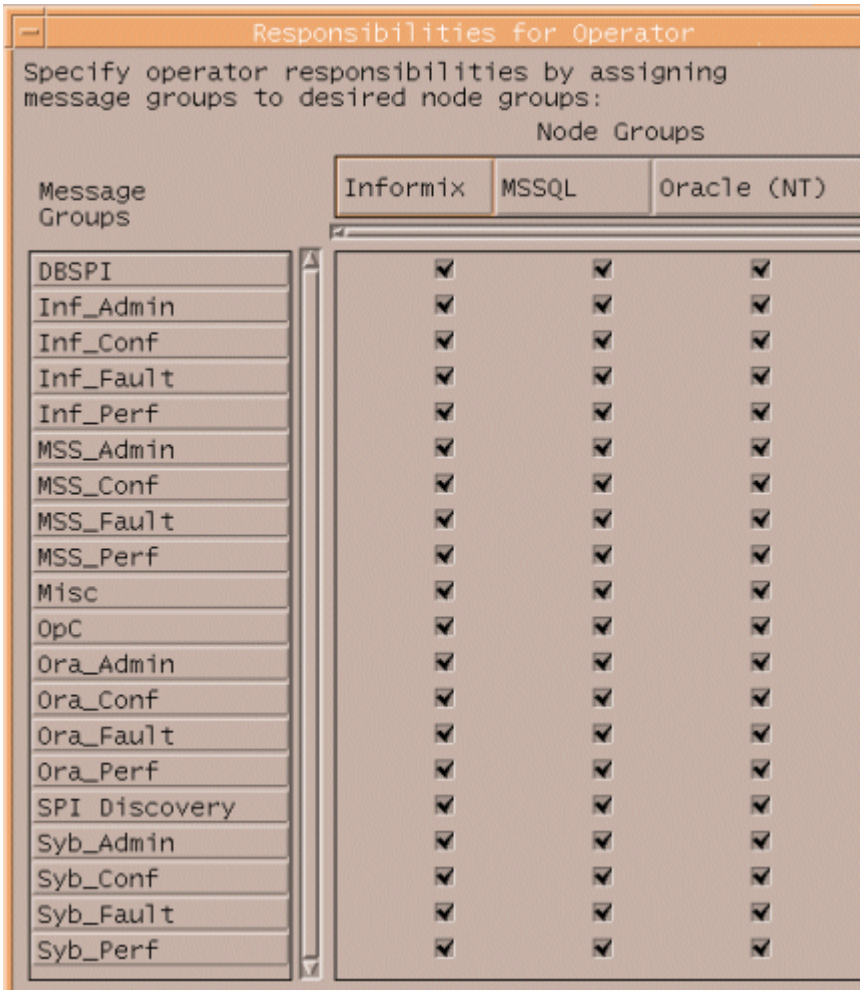
1 Log on to OVO as the OVO administrator.

2 Open the User Bank window, click the **<***operator***>**, right-click **Modify**, and select **Responsibilities**.

3 In the Responsibilities for Operator (`opc_adm`) window select DBSPI and the appropriate database-type for each managed node group as illustrated. Select `DBSPI` and the appropriate message groups from those listed below.

```
Inf_Admin, Inf_Conf, Inf_Fault, Inf_Perf
MSS_Admin, MSS_Conf, MSS_Fault, MSS_Perf
Ora_Admin, Ora_Conf, Ora_Fault, Ora_Perf
Syb_Admin, Syb_Conf, Syb_Fault, Syb_Perf
```

Figure 7 Assigning message groups to node groups Informix, MSSQL, and Oracle (NT)



4 Click **Close** to complete.

Task 7: Specify operators for Service Map viewing

If you installed **Service Navigator** (which accompanies OVO for UNIX), the following task is relevant.

The OpenView Operations Service Map must be assigned to an operator before the operator can access it. By default, Service Map viewing is automatically assigned to the *opc_adm* operator. You can modify the default setup by adding or removing any operator(s) from the Service Map contained in any of the five files (for each database type).

1 On the management server open a terminal window and log in as `root`.

2 Open the following directory by entering:
`cd /opt/OV/SPISvcDisc/conf`

3 Using the text editor of your choice, open one or each of the following five files:

- DBSPI_ORACLE_UNIX_DiscConfig.sh
- DBSPI_ORACLE_WIN_DiscConfig.sh
- DBSPI_SYBASE_DiscConfig.sh
- DBSPI_INFORMIX_DiscConfig.sh
- DBSPI_MSSQL_DiscConfig.sh

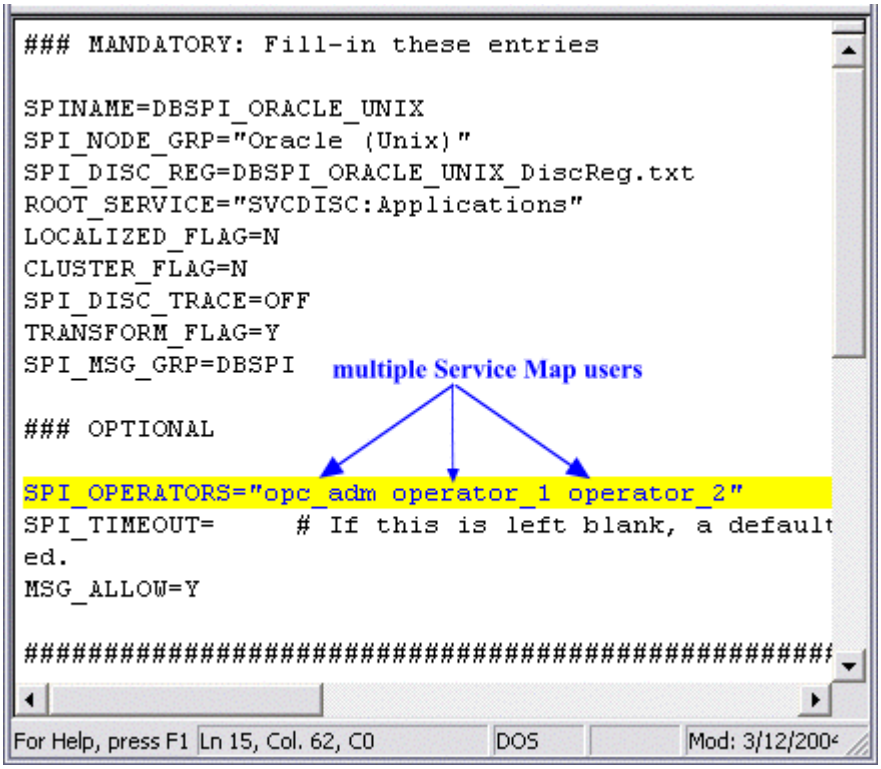
4 In the `SPI-Specific values, ###OPTIONAL` section, locate the line:
`SPI_OPERATORS=opc_adm` (see illustration on following page).

5 Insert a space after each new entry, and enclose the entire defined operators in double quotation marks.
For example, adding **operator_1** and **operator_2** would look like:

```
SPI_OPERATORS="opc_adm operator_1 operator_2"
```

6 Save and close the file.

Figure 8 You can add or remove users (leaving a blank space in between each) to allow or disallow views of the OpenView Operations Service Map



Task 8: Run Discovery

Running the Service Discovery application can assist you in the database connection configuration. This application, once run, populates the configuration file with some keyword entries.

Service discovery automatically occurs every night around 2:00 a.m., resulting in daily updates to the Service Navigator. For the initial service discovery and whenever you want an immediate update, use the DB-SPI Service Discovery application as described below.

1. At the OVO console from the Window menu select **Node Group Bank**.
2. From the Window menu select **Application Bank**.
3. In the Application Bank window double-click the **DB-SPI** application group and among the applications appearing in the window, double-click the **Service Discovery** application group.
4. One at a time, drag and drop the appropriate Database SPI node group to the corresponding service discovery application.

Node Group	Discovery Application
Informix →	Informix
MSSQL →	MSSQL
Oracle (NT) →	Oracle Win
Oracle (UNIX)→	Oracle Unix
Sybase→	Sybase

Wait until discovery completes for one group before moving to the next.

Figure 9 Dragging the node group onto the appropriate Service Discovery application starts the discovery process, which discovers database instances and partially configures the DB-SPI connection to those instances

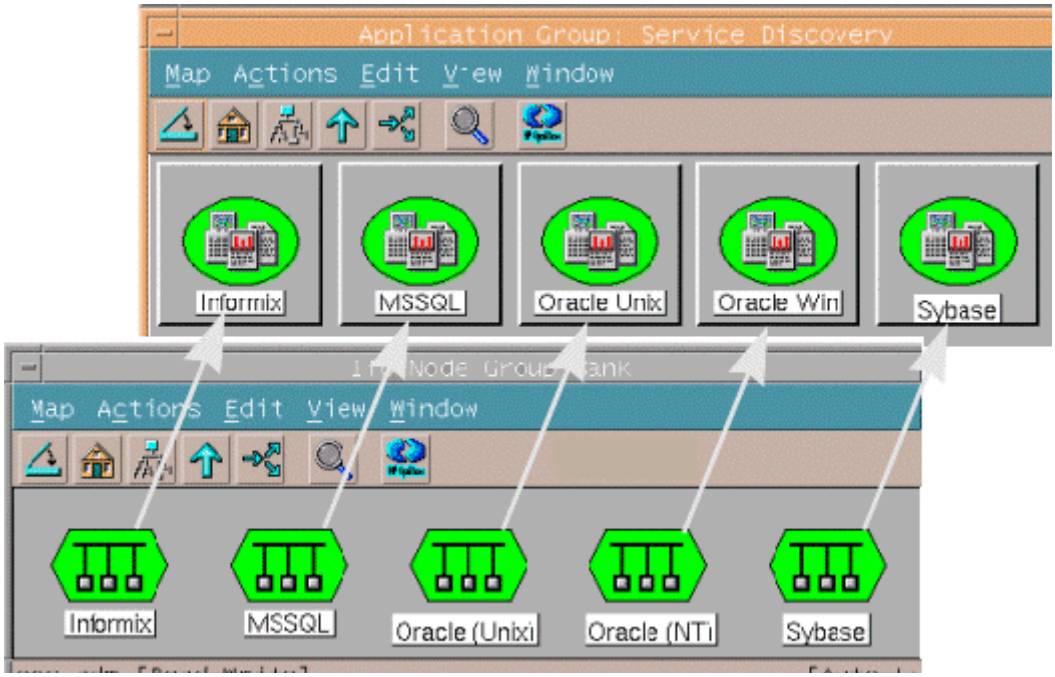
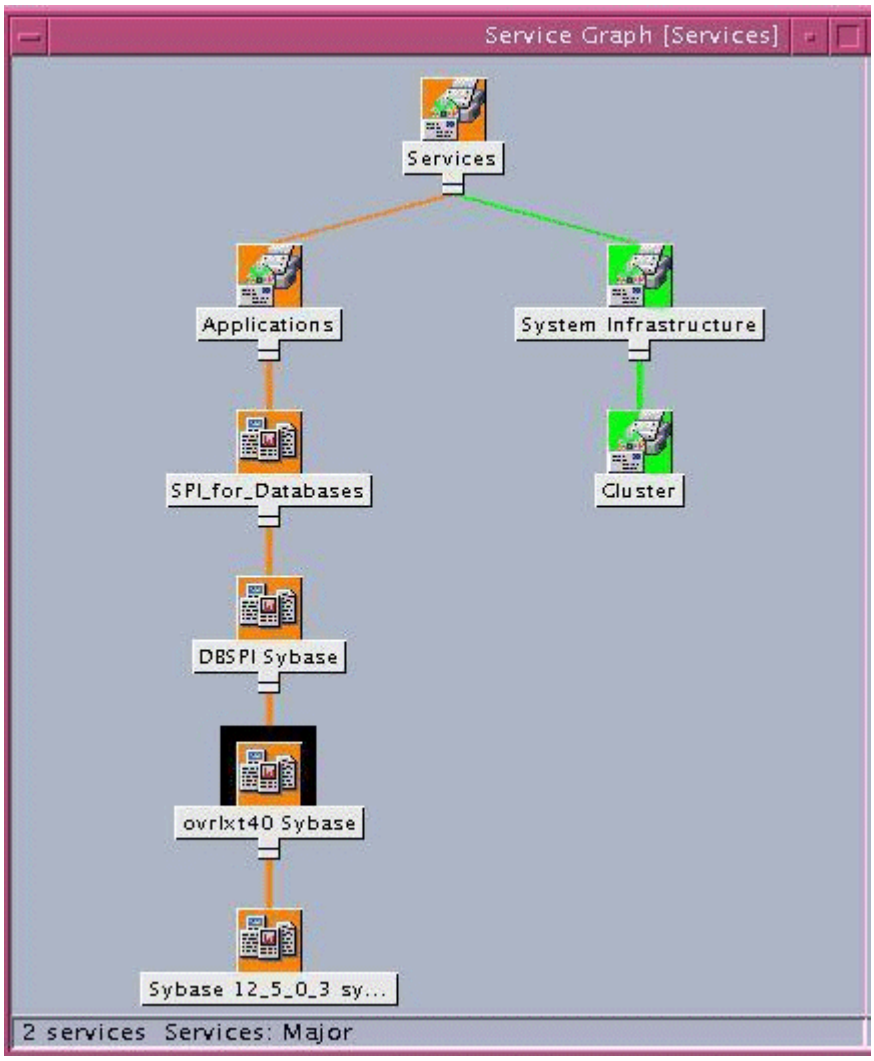


Figure 10 Service Navigator shows discovered database instances and can indicate service problems by changing the instance color



Task 9: Configure the database connection and (if appropriate) enable graphing/reporting

Essential to the Database SPI gaining access to the managed node database instances is the configuration file. This file is a text file that appears in your default text editor when you run the Configure DB Connections application against a selected managed node. For Oracle/ Windows and MS SQL Server, always use the Admin Windows→Configure DB Connections application.

Appendix C, Configuring Database Connections contains configuration file details and the required entries.

The table below shows the information that the Service Discovery application can access and insert in the configuration file. You must supply those configuration entries not provided through discovery:

Table 3 Service Discovery for each database type

DB Type	Platform	Discovery method	Config file entries	To configure	To add:
Informix	UNIX	user 'informix' & Informix file '`.profile`' in <*login*> directory	Home, Instance(s); Disabled	<*user*>/<*password*>; <*home*>; <*instance*>; delete the word "`Disabled`" next to the instance.	Instances not discovered
MS SQL Server	Windows	Registry	Instance(s); with Windows authentication	With Windows authentication, nothing. With standard security, add <*user*>/<*password*>	Should discover all instances
Oracle 8.x & 9.x, 10	Windows	Registry	Home, Instance(s) Disabled	<*user*>/<*password*>; delete the word "`Disabled`" next to the instance.	Should discover all instances
Oracle 10 ASM	Windows	Registry	Home, Instance(s) Disabled	Sys/<*sys_password*> delete the word "`Disabled`" next to the instance.	Should discover all instances
Oracle 7.x	Windows	No discovery	Nothing	<*home*>, <*instance*>, <*user*>/<*password*>	All Oracle 7.x instances
Oracle, all versions	UNIX	Oracle file `oratab*`	Home, Instance(s); Disabled	<*user*>/<*password*>; delete the word "`Disabled`" next to the instance name.	Instances not "discovered" in `oratab`
Oracle 10 ASM	UNIX	Oracle file `oratab*`	Home, Instance(s); Disabled	`CONNECT "/";` delete the word "`Disabled`" next to the instance name.	Instances not "discovered" in `oratab`
OEM 10g	Windows	Registry	Nothing	Nothing	Should discover OEM 10g

Table 3 Service Discovery for each database type

DB Type	Platform	Discovery method	Config file entries	To configure	To add:
OEM 10g	UNIX	Oracle file `oratab*`	Nothing	Nothing	Should discover OEM 10g
*On some UNIX systems, `oratab` is in an alternate location. In such situations you must perform a link so that DB-SPI can find the file in `/etc/oratab` or `/var/opt/oratab`. For example: `ln -s <path>/oratab /etc/oratab`.					
Sybase	UNIX	user 'sybase' & Sybase file 'dataserver' in <*login*> directory	Home, Instance(s); Disabled	<*user*>/<*password*>; delete the word "`Disabled`" next to the instance.	Instances not discovered

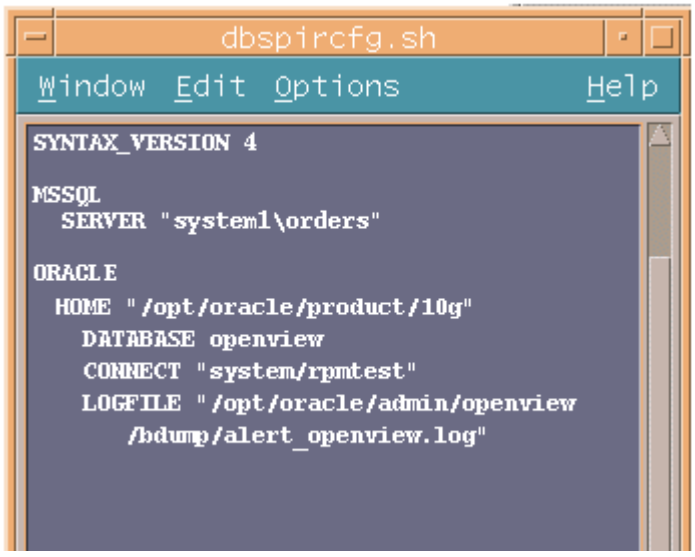
1 From the Window menu open the **Node Bank** and highlight the node group that you want to configure.

2 From the Window menu, select **Application Bank**.

3 In the Application Bank: window:
for UNIX nodes select **DB-SPI→Admin** group, or
for Windows nodes **DB-SPI→Admin Windows**.

4 Double-click **Configure DB Connections** to configure the selected managed node (or drag and drop the managed node onto the application).

Figure 11 The DB-SPI configuration file appears in a window used by your default text editor. See Appendix C, Configuring Database Connections for the correct syntax for entries in the file



5 Prepare a list of all **RDBMS** instances on the node with keywords and entries as shown in Configuration File Syntax on page 267.

6 Apply the configuration to the node by accepting **YES** to the prompt:

Save configuration to `/var/opt/OV/dbspi/local.cfg? [yes]`

7 If your entries are accurate, you receive the message:
`Verification was successful.`

8 To use OpenView Performance Manager, or PerfView (graphing tools), and/or Reporter, enable the product(s) by accepting [yes] to:
`Do you want to enable graphing and reporting data collections now? [yes]`

If you choose [no], you can later enable graphing and/or reporting by dragging/dropping the managed node on the Enable Graphs or Enable Reports application (in the *DBSPI→Admin [Windows]* group).

Task 10: Added configuration for OVO 8.x DCE Nodes

Because of new security restrictions in OVO 8.x, Database SPI graphing metrics need a special configuration in order to generate graphs through an operator action (selecting **Perform Action** in the Message Details of an alarm situation). Without the configuration, **Perform Action** is inactive.To allow operator actions to occur from all non-HTTPS nodes that have been added to the Database SPI node groups:

1 At the management server, edit the file:
`/etc/opt/OV/share/conf/OpC/mgmt_sv/remactconf.xml`

The file must appear similar to the following:

```
<?xml version="1.0"?>
<config xmlns="http://openview.hp.com/xmlns/Act/Config
/2002/08">


<!--
*****************************************************************************
The following active rule allows (enables) all remote actions from an HTTPS
node. Remote actions from DCE nodes are disabled.
*****************************************************************************
 -->
  <rule>
    <doc>Allow ALL certified actions</doc>
    <allow/>
  </rule>


<!--
*****************************************************************************
SPI discovery requirement: Allow (enable) remote actions from all non-HTTPS
nodes in the Oracle (NT) node group to the management server
*****************************************************************************
-->


  <rule>
    <doc>Actions from Oracle NT node group to management server</doc>
```

```
    <if>
      <source>
        <nodegroup>Oracle (NT)</nodegroup>
      </source>
      <target>
        <mgmtsrv/>
      </target>
      <certified>false</certified>
    </if>
    <allow/>
  </rule>



<!--
**************************************************************************
SPI discovery requirement: Allow remote actions from all non-HTTPS nodes in the Oracle
(UNIX) node group to the management server.
**************************************************************************************
 -->


  <rule>
    <doc>Actions from Oracle Unix node group to management server</doc>
    <if>
      <source>
        <nodegroup>Oracle (Unix)</nodegroup>
      </source>
      <target>
        <mgmtsrv/>
      </target>
      <certified>false</certified>
    </if>
    <allow/>
  </rule>



<!--
```

```
**************************************************************************
SPI discovery requirement: Allow remote actions from all non-HTTPS nodes in
the Sybase node group to the management server.
**************************************************************************
-->


  <rule>
    <doc>Actions from Sybase node group to management server</doc>
    <if>
      <source>
        <nodegroup>Sybase</nodegroup>
      </source>
      <target>
        <mgmtsrv/>
      </target>
      <certified>false</certified>
    </if>
    <allow/>
  </rule>



<!--
**************************************************************************
SPI discovery requirement: Allow remote actions from all non-HTTPS nodes in the Informix
node group to the management server.
**************************************************************************

 -->


  <rule>
    <doc>Actions from Informix node group to management server</doc>
    <if>
      <source>
        <nodegroup>Informix</nodegroup>
      </source>
      <target>
        <mgmtsrv/>
      </target>
      <certified>false</certified>
    </if>
```

```
    <allow/>
  </rule>
<!--
*****************************************************************************
SPI discovery requirement: Allow remote actions from all non-HTTPS nodes in the MSSQL
node group to the management server.
*****************************************************************************
 -->


  <rule>
    <doc>Actions from MSSQL node group to management server</doc>
    <if>
      <source>
<nodegroup>MSSQL</nodegroup>
      </source>
      <target>
        <mgmtsrv/>
      </target>
      <certified>false</certified>
    </if>
    <allow/>
  </rule>


</config>
```

2 Restart OVO server processes using the following command:
`opcsv -start`

Task 11: Run the Service Discovery application again (see Task 6)

You must run the service discovery application again to update the service map with newly configured (previously undiscovered) instances. This step is important to accurately populate the map with all new instances.

Task 12: Distribute DB-SPI templates

> It is recommended you use the Quick Start templates for initial setup. Later if you decide to customize template selection, please refer to the template compatibility matrices in Appendix B, Components.

The Quick Start templates, according to database type, are automatically assigned to the default node groups as shown in the table below; you need only distribute them.

Node Group	Quick Start Group Assigned
Informix	DBSPI-Informix: Quick Start
MSSQL*	DBSPI-MSS7: Quick Start
Oracle (NT)	DBSPI-Oracle: Quick Start (Windows)
Oracle (UNIX)	DBSPI-Oracle: Quick Start (UNIX)
Sybase	DBSPI-Sybase: Quick Start

*If you use MS SQL Server 6.5, you must manually create a node group for the nodes running MS SQL Server 6.5. Then you can manually assign the *DBSPI-MSS6: Quick Start* template group to the group and use the procedure that follows to distribute the templates.

1 Open the OVO Node Group Bank, and select the group(s) to which you have added nodes to monitor from among the choices:
- Informix
- MSSQL (for MSSQL Server 7 and higher)
- Oracle (NT)
- Oracle (UNIX)
- Sybase

> For detailed instructions on OVO template assignment, please refer to the OVO online Help or the *OVO Concepts Guide*.

2 To distribute the (already) assigned templates, from the Actions menu select **Agents→Install/Update SW & Config**.

3 In the Install/Update OVO Software and Configuration window check the **Templates** check box.

4 Select the **Nodes in List** option button.

5 If you did not previously select a node, click the **Get Map Selections** button to list the target node(s) and click **OK** to distribute templates to the managed node(s).

The following message is displayed in the Message Browser:

```
The following configuration information was successfully distributed:
Templates
```

The DB-SPI templates are now distributed to the selected managed node(s). DB-SPI monitors can now begin running every 5 minutes, 15 minutes, 1 hour, and 1 day (according to template group selections).

Checking for Completeness of Installation

Before you begin using DB-SPI extensively, you can check your installation on the managed nodes to ensure that all filesets and program components are present.

Verifying DB-SPI Version Installed on the Managed Nodes

The Verify Deployment application displays the version information of installed components and verifies the database to help in troubleshooting DB-SPI on the managed node.

To run Verify Deployment, follow these steps:

1. Log on to OVO as the OVO Administrator.
2. Select the desired managed node.
3. Select **Window: Application Bank→DBSPI→Admin (or Admin Windows) →Verify Deployment**.

   The Verify Deployment window is displayed, showing information about the connection and the DBSPI files located on the managed node.
   Optionally, on UNIX run *<OVO commands directory>*`/dbspiverify`.
4. Press **Enter** to exit the window.

You have now completed the DB-SPI installation and configuration process. The following section provides information on removing DB-SPI. You need to complete this procedure when you install new versions of DB-SPI.

De-installation

This process completely removes the DB-SPI installation, including DB-SPI templates, from the OVO Management Server. Any customization (copies of DB-SPI default templates) residing in other OVO template groups should also be removed. Follow these steps in order to ensure a proper removal. *If you are upgrading DB-SPI, complete only the section "Removing Software from the Management Server."*

Starting cleanup on managed nodes

The Cleanup application removes runtime files like trace, log, and data storage. Remove these files prior to removing the Database SPI. Start the clean-up process as follows:

1. From the Window menu, open the **Application Bank→DB-SPI Admin**.
2. Open the Node Bank window, displaying those nodes with the DB-SPI installed.
3. Drag and drop each node onto the **Cleanup** application (select the node and execute the Cleanup application.

Removing software from the management server

1. Log on as `root`, and open a terminal window.

2 In the terminal window verify that the DISPLAY environment variable is set correctly to your workstation by entering:

echo $DISPLAY

If the DISPLAY environment variable is not set correctly, set it correctly.

3 In the terminal window start swremove by entering:

/usr/sbin/swremove

Or, you can run swremove from the command line by entering:

/usr/sbin/swremove DBSPI

Then skip ahead to step #7.

4 Select **DB-SPI** from the swremove main window and mark it for removal by selecting **Mark for Remove** from the Actions menu**.**

5 Start the deinstallation analysis by selecting **Remove (analysis)** from the Actions menu.

6 Select the **Logfile** button to review the results of the analysis. If the analysis phase succeeds, begin the removal as follows:

a Click **OK** in the Logfile window.

b Click **OK** in the Remove Analysis window.

c Click **Yes** in the Confirmation window.

If the analysis phase fails, follow the recommendations made in the logfile located at /var/adm/sw/swagent.log and run the analysis again before proceeding.

7 To ensure removal of all DB-SPI files from the management server area, delete any file in any directory below /var/opt/OV/share/databases/OpC/mgd_node/customer that begins with "dbspi."

The **swremove** command removes the files from the file system only. The DB-SPI templates are still in the OVO data repository and must be deleted manually. First, however, the DB-SPI templates (and the DB-SPI software) must be de-assigned from the managed nodes.

If you are upgrading the Database SPI, you must stop here and complete the tasks described in the section Required Tasks on page 21

Disabling graphs and reports

Perform this step if you use graphing and reporting tools with DB-SPI. Disabling graphs and reports on the managed nodes using the **Disable Graphs & Reports** application deletes entries in the perflbd.rc file (UNIX) or perflbd.mwc (Windows—MeasureWare only).

To disable the graphing and reporting:

1 Select the node on which you want to disable MeasureWare Agent integration (done in conjunction with disabling reports).

2 Open the Application Bank window and double-click the **DB-SPI→Admin** group.

3 Double-click **Disable Graphs & Reports**.

For MeasureWare Agent only: When convenient, you should restart the OpenView Performance Agent (MeasureWare Agent) after any changes by entering the command or completing the action:

(UNIX) `/opt/perf/bin/mwa restart server`
(Windows) Use the OpenView Performance Agent (MeasureWare Agent NT) GUI.

Removing template assignments from managed nodes

1. Open the OVO Node Bank window and select the targeted node.
2. To un-assign the templates from the node, select **Agents→Assign Templates** from the Actions menu.
3. In the Define Configuration window select all DB-SPI templates or template groups, including any user-customized templates.
4. Select **Remove Templates** and click **OK**.

Removing software from managed nodes

1. Select **Install/Update SW & Config** from the **Actions** menu.
2. Check the following check boxes to select these components for removal:

   - Templates
   - Actions
   - Monitors
   - Commands

3. Select the **Nodes in List** option button.
4. If you did not previously select a node, click the **Get Map Selections** button to list the target node(s).
5. Select the **Force Update** option button.
6. Click **OK** to remove the Templates, Actions, Commands and Monitors from the managed node(s). The following message is displayed in the Message Browser:

   The following configuration information was successfully distributed:
   Templates Actions Commands Monitors <node name>

Deleting templates, applications, node groups

You must manually delete DB-SPI templates, which are the Message Source, Message Group, and Application Group templates. To speed the process, use the SHIFT-click method to select multiple templates for deletion.

1. To delete the DB-SPI Message Source templates, from the Window menu select **Message Source Templates**.
2. Open all DB-SPI template groups and use the SHIFT-click method to select all templates for deletion.
3. Select **Delete from All...** to delete the templates.
4. Select **YES** in response to the message:

   Do you really want to delete the template(s)?

5 To delete the DBSPI Message Group and its contents, from the Window menu select **Actions→Message Group→Delete.**

6 Right-click **DBSPI** and select **Delete→From All Submaps**.
(For OVO 6.0, holding down the Ctl key, select each message group and use **Action→MsgGroups→Delete**.)

Ora_Admin, Ora_Conf, Ora_Fault, Ora_Perf or
Inf_Admin, Inf_Conf, Inf_Fault, Inf_Perf; or
MSS_Admin, MSS_Conf, MSS_Fault, MSS_Perf or
Syb_Admin, Syb_Conf, Syb _Fault, Syb_Perf

7 Click **OK** in response to the message:

```
About to delete selected objects.
```

8 To delete the DB-SPI application groups and their contents, from the Window menu, select **Application Bank**.

9 Right-click the DB-SPI application group and select **Delete**.
(or **Actions→Application→Delete**.)

10 Click **Yes** in response to:

Do you really want to delete the application group?

11 To delete DB-SPI node groups, from the Window menu select **Node Group Bank**.

12 In the Node Group Bank window right-click the DB-SPI node group(s) (up to five node groups may have been created automatically, depending on product selection), and select **Delete**. (or **Actions→Application→Delete**.)

13 Click **Yes** in response to:

Do you really want to delete the node group?

Finishing Cleanup on Managed Nodes

Because some programs may have continued running after you completed running the Cleanup application at the beginning of the de-installation procedure, you may want to clear away whatever nominal data may remain.

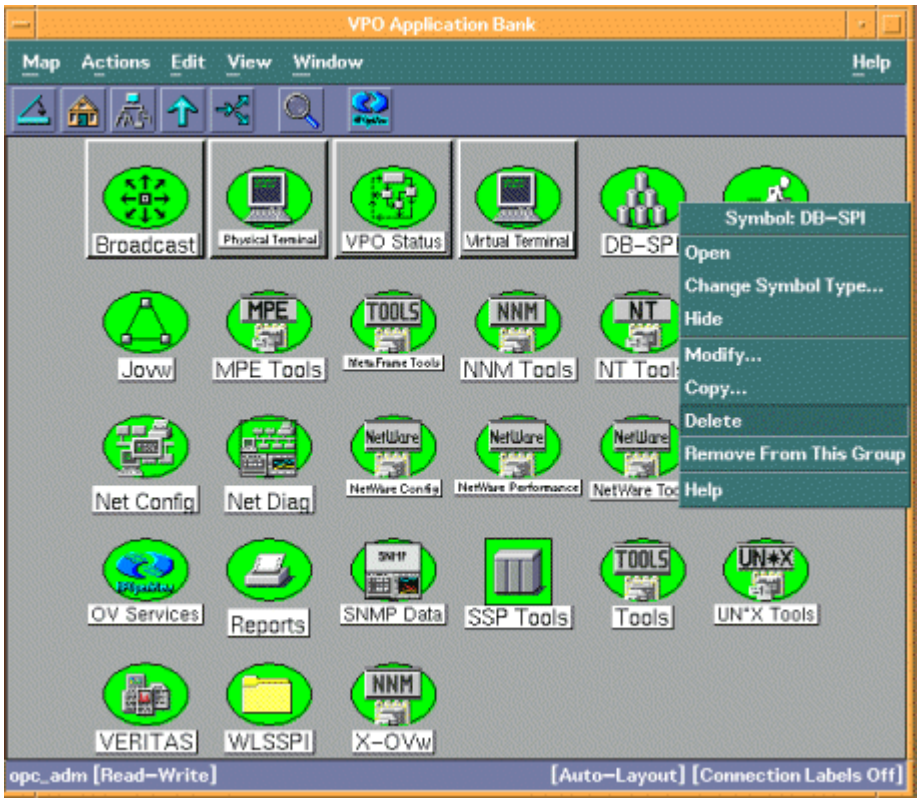
To completely remove any data files remaining, delete these directories and their contents from each managed node:

Table 4 Removing Remaining Software from Managed Nodes

Managed Node OS	path/directory
HP-UX, Linux, Solaris, Tru64	/var/opt/OV/dbspi
AIX DCE AIX HTTPS	/var/lpp/OV/dbspi /var/opt/OV

You have now completed the DB-SPI deinstallation process.

Figure 12 Deleting the DB-SPI Application Groups



3 Using and Customizing the Database SPI

Service discovery: The Smart Plug-ins for Databases (DB-SPI) integrates with the OVO Service Navigator, showing the topology of your database installations and database availability/performance service alerts as needed. Integration with Service Navigator requires only that you install the Discovery template on the management server, which runs the discovery program every night at 2:00 a.m.

Service monitoring: The DB-SPI also includes a template group, Quick Start, that makes it easy for you start monitoring the discovered database services. After you have deployed Quick Start, DB-SPI can start retrieving/interpreting data immediately. As a result, any service alerts/problems can then be communicated to the OVO Service Navigator and the OVO Message Browser.

This chapter provides detail on the templates for purposes of editing/customizing them, and focuses on those areas that you might want to modify. The following topics are covered:

Using DB-SPI Templates

Although the OVO templates can be customized, they require very little customization when DB-SPI is installed.

This section covers the following topics:

- Template Groups
- Roll-up & Drill-down Metrics

Template Groups

DB-SPI template groups organize how you deploy message-, graph-, and report-generating components. Among the DB-SPI template groups are metric, message, logfile, and Reporter templates. The template set of major importance to you is **Quick Start**, which you can see in the Message Source Templates window under the DBSPI-<*database_application*> appropriate to the database type you are using.

The Quick Start template set provides you with a simple means of deploying all the templates you need to monitor your database application right away. The template groups within the Quick Start set are similar, although not identical, for Oracle, Sybase, Informix, and MS SQL Server. Each Quick Start template group set includes:

— **Standard** or **Favorites**: for messages generated from database performance and event metrics grouped according to collection intervals and data collections sent to the separately purchased OpenView Reporter or OpenView Performance Manager.

— **Messages**: for messages generated from DB-SPI programs (can inform you of errors in programs executing or making database connections).

— **Reporter**: for reporting capabilities with the separately purchased HP OpenView Reporter.

Of these template groups, the Favorites (for Oracle) or Standard (for Informix, Sybase, and MS SQL Server) groups are those that generate alarms, based on the performance and availability of your database application. These groups contain *monitor* and *collector* templates that you may want to change in terms of the threshold value settings for generating the alerts/alarms (messages) displayed in the OVO Message Browser. For this reason these template types are discussed in the sections that follow.

Monitor Templates and Collector Templates

Within each database Quick Start: Favorites or Standard group, you see a list of monitor templates and collector templates.

— *Monitor templates* pertain only to individual metrics.

— *Collector templates* pertain to all metrics collected in the specified collection interval.

Monitor templates define how data is collected for the individual metric and set a threshold value that when exceeded generates severity-level alerts/messages in the OVO Message Browser. In OVO, you can change the threshold within a monitor template by highlighting the template and selecting the **Conditions...**button.

Collector templates define all metrics for the database application that are collected in the specified collection interval. You can see how this works by opening a collector template (always identified by a collection interval). In this template the list of targeted metrics (by number) follow the collector/analyzer command (**dbspicao | dbspicai | dbspicas | dbspicam**) within the `Command` text box in OVO.

Roll-up & Drill-down Metrics

Roll-up metrics address those situations where one metric value has multiple occurrences within database instances. The drill-down metric allows you to see a specific occurrence of the roll-up metric. Even though all database applications have roll-up and drill-down metrics, the settings for roll-up metrics are handled differently for Oracle.

In Oracle, you set thresholds in both the collector template and the monitor template conditions. In Sybase, Informix, and MS SQL Server, you set thresholds only in the monitor template conditions. For this reason, Oracle metrics are covered in Customizing Oracle Roll-up Metrics on page 105, while Sybase, Informix, and MS SQL Server metrics are covered in Changing Thresholds for Monitor Templates on page 56.

The reports for these metrics provide details about the over-threshold objects if desired. In addition, the corresponding Sybase, Informix, and MS SQL Server drill-down metrics can be used to provide additional detail.

Drill-down metrics for Sybase, Informix, and MS SQL Server use the **object** field which includes the name of the server as in: <*server*>`:object` (rather than the database name as in Oracle: <*database*>`:object`).

The **object** name in the object field which is "<*server*>:object" is sent to OVO to allow thresholding for specific objects (for Oracle, see Chapter 5, Oracle Advanced Customizations).

Basic Template Customizations

After you begin using DB-SPI, you may decide that specific DB-SPI templates need some modification. To customize any template, you simply open it and modify as necessary. For these kinds of customizations, you would likely change the default templates on the OVO Management Server and then re-deploy them to the managed nodes. Basic customizations, it is assumed, would be implemented for every managed node.

Advanced customizations are generally more involved and made for specific systems. In some cases it is advisable to make copies of the original templates because your new templates are for specific areas of database management that are not likely repeated across the organization.

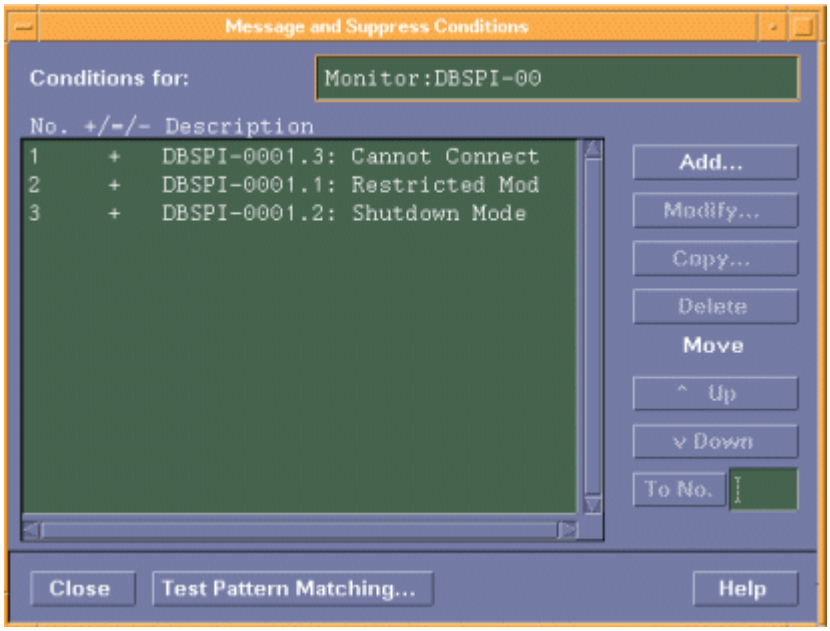
Modify Monitor Templates For All Databases

Many metric attributes can be easily modified for all databases being monitored by following these steps.

1. Select the **Message Source Templates** window.
2. Open the template group containing the metric to modify.
3. Double-click the desired metric to open the Message and Suppress Conditions window.
4. Select the condition to modify (there is usually only one).

5 Click the **Modify** button.

Figure 13 The Message and Suppress Conditions Window for Oracle metric 1, showing three defined conditions



The Condition window is displayed. The following attributes can be easily modified:

— **Threshold.** Set the desired threshold. Sybase, Informix, and MS SQL Server metrics that can be customized in this way are listed in the *Sybase Reference*, *Informix Reference*, and *MS SQL Server Reference* guides respectively. Oracle metrics are addressed in Chapter 5. The Default Threshold identifies metrics in which the OVO threshold can be modified. The OVO threshold for Oracle "roll-up" metrics (Oracle 6, 8, 9, 11, 17, 18, 31, 38, 69) should not be changed. The DB-SPI Metric Parameter is used to modify the alarming of such metrics (see page 49). The OVO threshold for metrics that return a count (Oracle metrics 4-5, 7, 16, 61, 63, 67-69, 77-81) should also not be modified.

— **Message Text**. Be careful not to modify any of the parameters in a message. Parameters are surrounded by <> brackets and begin with $.

— **Duration.** Most metrics are defined as Message Generate type Without Reset and without a Duration. Please consult the *OVO Concepts Guide* or Help for more modifying this field.

— **Severity.** Click the **Severity** button and select the desired severity setting.

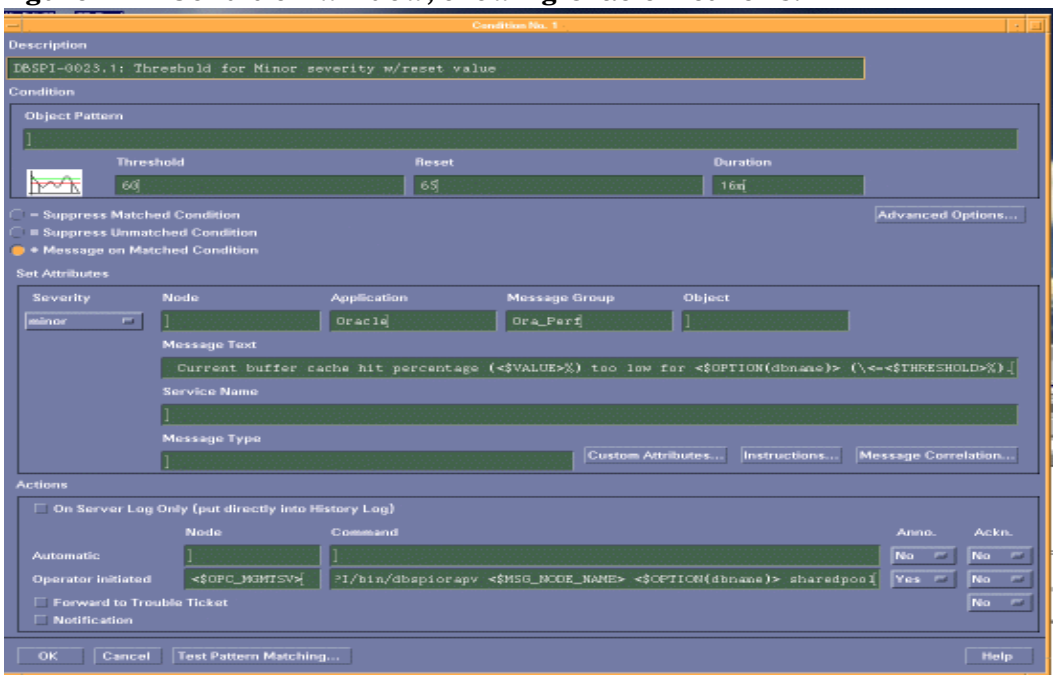
— **Actions.** This field provides the ability to add custom programs, use **SQL* Plus, ISQL, DBAccess** or other custom applications to generate complex reports, etc. Two types of actions are available:

Operator initiated. These actions are performed only upon the initiation of an operator.

Automatic. These actions are performed automatically when the metric alarms.

Most metrics have operator-initiated actions which when chosen display a PerfView or OVPM graph (if PerfView is installed on the management servers or OVPM is configured). Some metrics also have an automatic action which runs a report.

Figure 14 A Condition Window, showing Oracle metric 23.1



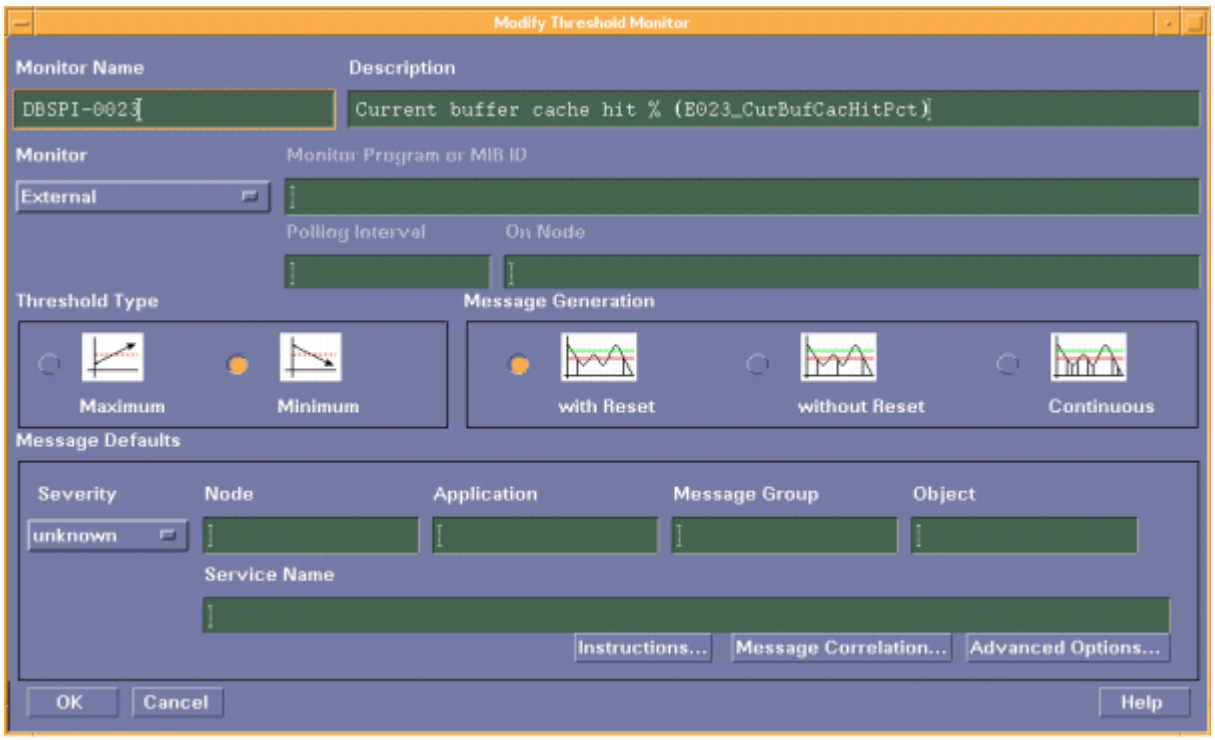
An alarm can be generated once or multiple times, depending on its setting, which can be as follows:

- **Reset:** Alarms are generated once when the threshold value is exceeded. At the same time a reset threshold value is activated. Only when the reset threshold value is exceeded, does the original threshold value become active again. Then when the threshold value is again exceeded, another alarm is generated and the process starts all over again.
- **Without Reset.** Alarms are generated once when the monitoring threshold value is exceeded. Alarms reset automatically when metric values are no longer in violation of the thresholds and are generated again when the threshold is exceeded.
- **Continuously**. Alarms are generated continuously when metrics exceed the threshold.

To change the Alarm Generation Type (usually in conjunction with the Threshold, Polling Interval and Reset value) follow these steps:

1. Open the **Message Source Templates** window.
2. Double-click the template group that contains the monitor template (metric) you want to modify.
3. Click the **Modify** button.
4. Click the button next to the desired Message Generation type.

Figure 15 A Modify Threshold Monitor Window showing Oracle metric 23



Advanced Template Customizations

Suggestions follow for changes you might make in customizing templates and database metrics. The suggestions here range from making copies of default template groups in order to customize a few settings, to deleting whole groups of metrics within a template's command line.

This section is considered *advanced*, despite the simplicity of implementing many of the suggested changes, as some advanced database metric knowledge is often required in order to determine the necessary modifications.

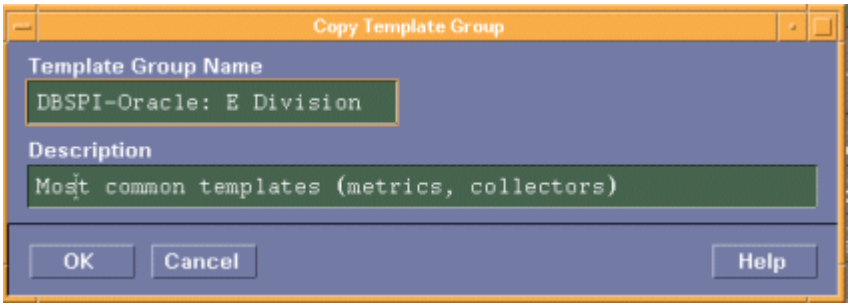
Creating a New Group with Chosen Metrics

Before beginning customization, first decide which metrics you are interested in without touching or changing the thresholds. It is recommended that you make a copy of the original group and use the copy for customization purposes.

As a start, determine which metrics you want to change and what templates within the group you want to use. Then proceed as follows:

1. Open the Message Source Template window.
2. Highlight the group you want to use and select the **Copy...** button.
3. Rename and save the group.

Figure 16 Creating a new template group (OVO only) by copying, modifying an existing one



4 Likewise, within the renamed template group, copy each original template and rename it.

5 Alter the renamed templates within the group as necessary.

Using the DB-SPI Collector/Analyzer Command with Parameters

The `dbspica<`*database_application_initial_character*`>` command is included in every collector template for each of the “favorites” groups designated according to collection interval. You can view the default command line parameters within each collector template in the **Command** text box in OVO.

The DB-SPI collector/analyzer starts data collection with the command:

- For **Oracle**: `dbspicao`
- For **Sybase**: `dbspicas`
- For **Informix**: `dbspicai`
- For **MS SQL Server**: `dbspicam`

With command `dbspicao` (Oracle), `dbspicas` (Sybase), `dbspicai` (Informix), or `dbspicam` (MS SQL Server) you can include the following parameters:

Table 5 Collector Command Parameters

Parameter	Function	Syntax
`-c`	(**collect**) Specifies template group that contains metric(s) on which to collect data. The DBSPI-Inf-Daily collector does not use the -c option. To avoid error messages in the `opcerror` log, use an existing monitor name for the -c option of any scheduled actions.	`-c <template_group_name>`
`-m`	(**metric**) Specifies the metric numbers or number ranges on which to collect data.	`-m <metric_number,metric_number_range>`
`-t`	(**tag**) Allows you to create a new template group by adding a prefix to an existing monitor template along with the metric number.	`-t <new_template_group_name>` -m `<metric_number>`
`-v`	(**verbose**) Shows additional detail	`-v (command line only; do not use in collector template)`
`-i`	(**instance**) Specifies the database instance (optional)	`-i <database_instance_name>`
`-e`	(**exclude**) Specifies database instance to exclude	`-e <database_instance_name>`
`-r`	(**report**) Shows metric values for the specified metrics.	`-r <metric_number_range>`
`-p`	(**print**) Prints the specified metrics and their values to standard output.	-p `<metric_number>` `<metric_number_range>`
`-d`	(**database**) Checks the connection to the database configured in the `local.cfg` file.	`-d` (command line only; do not use in collector template)

Table 5 Collector Command Parameters

Parameter	Function	Syntax
`-l`	(**logfile path**) Prints the database logfile paths to standard output.	`-l` (command line only; do not use in collector template)
`-n`	Attempts to connect to each database (or only one if the -i option is used) using SQL*Net; Oracle only.	`dbspicao -n -v`
`-w (Only for Sybase)`	Provides the number of seconds to collect Sybase metrics using DBCC commands (the equivalent of sp_sysmon). By default, **dbspicas** collects for 120 seconds (2 minutes) and the template launches **dbspicas** every 5 minutes (it collects for 2 minutes, rests for 3 minutes, collects for 2 minutes).	`-w <seconds>`

Syntax Examples (showing required order of parameters):

- To specify metrics to collect:

  `dbspicao -c <monitor_template_name> -m <metric_number_range>, <metric_number>`

  `dbspicao -c DBSPI-Ora-15min -m 6-9,11`

- To create a custom metric collection (OVO only) using the tag feature:

  `dbspicao -c SAP-DBSPI-Ora-15min -m 16 -t SAP-`

- To differentiate database instances:

  `dbspicao -c SAP-DBSPI-Ora-15min -m 203,16:1,17:80,18:90,58,62-65,77-81 -i DB2 -i DB3`

(Inserting “DB2” and “DB3” in the Command text box of the collector template results in collecting data for the specified metrics from these database instances only).

- To print a report:

  `dbspicao -r <report_number> -m <metric_number_range> -i <database_instance_name>`

  `dbspicao -r 1 -m 17,18`

- To specify greater detail be included in the output:

  `-v` (verbose) produces more verbose output (depending on the function).

- To print metric values for range of metrics for a database instance:

  `dbspicao -p -m range [-v] [-i <database_instance_name>]`

  `dbspicao -p -m 1-89`

- To check the database connection:

`dbspicao -d [-f] [-v] [-i <database_instance_name>]`

`-d` (database) performs a connection check to the databases configured in the `local.cfg`.

`-f` (filter) checks each filter defined for the database instance, for example:

`dbspicao -d -f -v`

The following command line option is for Sybase **(dbspicas)** only.

- To change the Sybase collection interval using `-w secs`:

  This option provides the number of seconds to collect Sybase metrics using DBCC commands (the equivalent of sp_sysmon). By default, **dbspicas** collects for 120 seconds (2 minutes) and the template launches **dbspicas** every 5 minutes (it collects for 2 minutes, rests for 3 minutes, collects for 2 minutes).

Accessing Drill-down Metric Data

To receive drill-down metric data you need to assign the drill-down template group to the managed nodes. To modify a condition using the `condition:object` pattern can be complex. Please refer to OVO Help for how to construct syntax for modifying or creating new metric conditions.

1. After installing DB-SPI, you can see the template groups
   DBSPI-Informix Drill Down
   or
   DBSPI-Sybase: Standard
2. Before deploying the template group, you can make changes such as adding a condition to a metric to monitor specific databases and objects or vary the threshold by database and object of these templates (see example that follows).
3. For Informix assign **DBSPI-Informix: Drill Down.** For Sybase assign **DBSPI-Sybase: Standard**.
4. Distribute templates to all appropriate nodes.

Example: Customizing Drill-down Object Conditions

Customizing the metrics to vary the threshold by object in OVO requires an understanding of how OVO handles metric condition matches. To analyze metric conditions, OVO starts at the first condition and works down until it finds a condition that qualifies, or runs out of conditions. For syntax example, see Table 29 on page 106. This example is relevant to Sybase and Informix, even though it appears to apply only to Oracle.

To customize the threshold for individual databases and objects:

1. Create a custom condition for each object.
2. Preceding the “catch all” condition, create a “**-Suppress Matched Condition**” so that OVO continues processing condition matches and does not stop at the “catch all” condition.

Changing Collection Intervals for Collector Templates

To change the metric collection interval, change the `Schedule` in the appropriate collector template. For example, to change the collection of default metrics from 5 minutes to 10 minutes for the Oracle Favorites template group, follow these steps:

1. Select the **Message Source Templates** window.
2. In the template group **DBSPI-Oracle**, open **DBSPI-Oracle: Quick Start→DBSPI-Oracle Favorites**.
3. Select the template **DBSPI-Ora-05min-Favorites**.
4. Click the **Modify...** button.
5. Change the **Scheduled Action Name** to **DBSPI-Ora-10min-Favorites**.
6. Change the **Schedule** to 0,10,20,30,40,50,60,70,80,90.

Only in the case of Sybase, use the `-w` command line option to change the schedule. For more information about the option, see Table 5 on page 52.

7. Modify the "-c" option of the command line to reflect the new metric name `(DBSPI-Ora-10min-Favorites)` as follows:

   ```
   dbspicao -c DBSPI-Ora-10min-Favorites...
   ```

8. Distribute the new **DBSPI-Ora-10min-Favorites** template as in Distribute DB-SPI templates on page 38.

Changing Collection Intervals for Selected Metrics

To change the metric collection interval for selected metrics, copy the appropriate collector template and change the `Schedule`. For example, to change the collection interval to 10 minutes for some Oracle metrics currently being collected at 5 minutes, follow these steps:

1. Select the **Message Source Templates** window.
2. In the template group **DBSPI-Oracle**, open **DBSPI-Oracle: Quick Start→DBSPI-Oracle Favorites**.
3. Select the template **DBSPI-Ora-05min-Favorites**.
4. Click the **Copy...** button.
5. Change the **Scheduled Action Name** to **DBSPI-Ora-10min-Favorites**.
6. Change the **Schedule** to 0,10,20,30,40,50,60,70,80,90.

Only in the case of Sybase, use the `-w` command line option to change the schedule. For more information about the option, see Table 5 on page 52.

7. Modify the "-c" option of the command line to reflect the new metric name `(DBSPI-Ora-10min-Favorites)` as follows:

   ```
   dbspicao -c DBSPI-Ora-10min-Favorites...
   ```

8. Delete the new 10-minute metrics from the **DBSPI-Ora-05min-Favorites** template. From the `Command` text box, remove only those metric numbers that should now be collected every 10 minutes. For example:

   ```
   dbspicao -c DBSPI-Ora-05min-Favorites -m 1-2,7,11:50,14,54,59-60,67,69:5
   ```

9 Delete 5-minute metrics from the new **DBSPI-Ora-10min-Favorites** template. Since this is a copy of DBSPI-Ora-05min-Favorites, from the `Command` text box, remove the metric numbers that will continue to be collected at 5-minute intervals (they will reside exclusively in DBSPI-Ora-05min-Favorites). For example:

```
dbspicao -c DBSPI-Ora-10min-Favorites -m 20-24,26-30,31:95,32-35,37,38:2
```

10 Distribute the new **DBSPI-Ora-10min-Favorites** and **DBSPI-Ora-05min-Favorites** templates as described in Distribute DB-SPI templates on page 38.

Changing Thresholds for Monitor Templates

Customize the threshold as needed (using non-"roll-up" metrics only: Oracle metrics 19-24, 26-28, 32-35, 39-40, 42-43, 45-46, 48, 50, 52, 54, 56-60, 62, 64-66, 75, 82-83, 85, 87, 89). For example, you may want to set the threshold for SAP databases for metric 54 to 10 minutes and leave it 20 minutes for all other databases. To do so, copy the existing condition to use as the default, and modify the original condition to serve as the exception. Follow these steps:

1 Double-click to open the metric to customize (**DBSPI-0054**).

The Message and Suppress Conditions window is displayed.

2 Select the desired condition.

3 Press the **Copy...** button to make a copy of the condition.

4 Name the condition **DBSPI-0054.2** and press **OK.** (This becomes the default condition.)

5 Select the original condition DBSPI-0054.1.

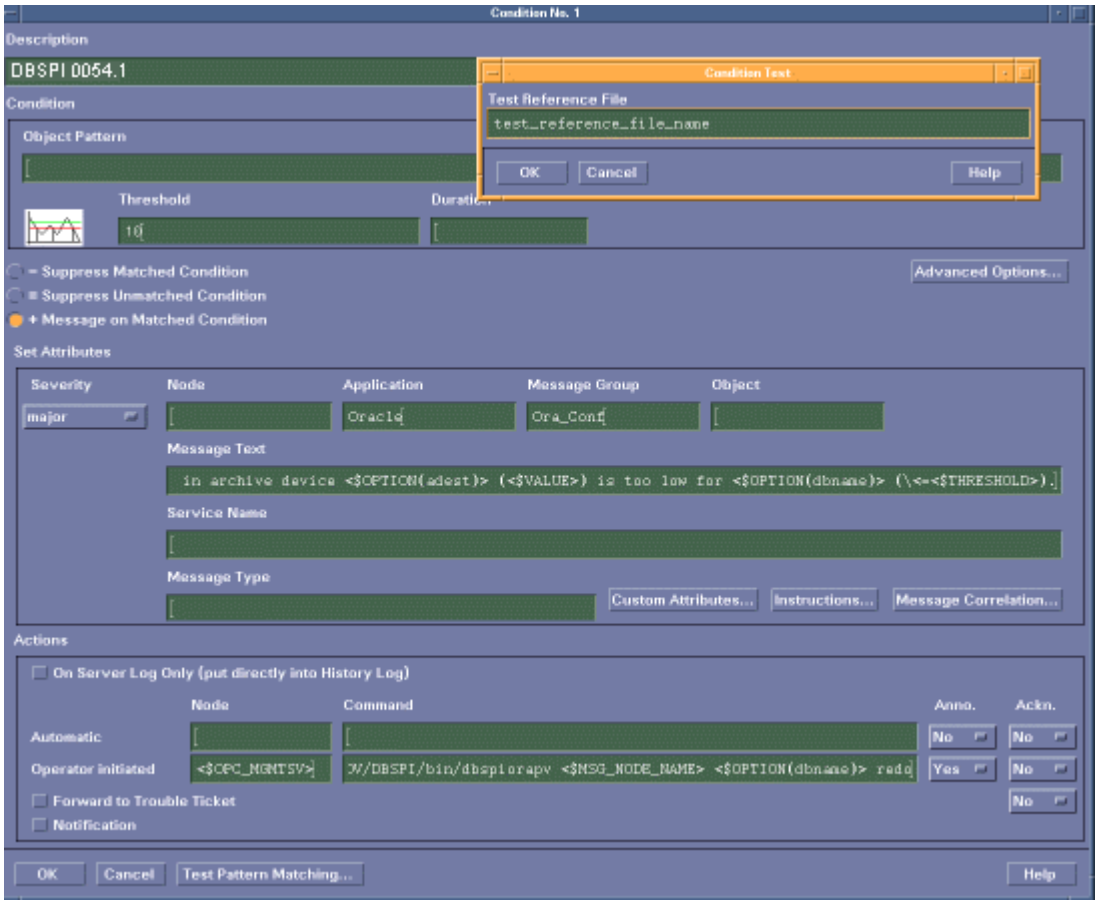
6 Press the **Modify** button to display the Condition window.

7 In the **Object Pattern** field, enter the desired characters to use for pattern matching. For example, to match on all databases that have SAP anywhere in the database name, enter SAP in the `Object Pattern` field. (see the OVO on-line help or the *OVO Administrators Task Guide* for pattern matching syntax).

8 Click the **Test Pattern Matching...** button to test the pattern and verify pattern matching (you must set up a match file first).

9 Change the value in the Threshold field from 20 to **10**.

Figure 17 Testing Pattern Matching from the Condition Window



Changing Threshold for Oracle Roll-up Metrics (6, 8, 9, 11, 16, 17, 18, 31, 38, 69) for Different Databases

If you have more than five databases to monitor on a system, you may want to use the tag feature described in Creating Custom Templates for Large Scale Environments on page 59. If you have fewer than five databases, you may want to set up a separate collector template for each database for a specific metric. For example, you have five databases SAP01, SAP02, SAP03, SAP04 and SAP05, for which you need different settings for the number of available extents. For SAP01 you would like 5; for all others you would like 10. To change the setting, you would follow these steps:

1. Select the **Message Source Templates** window.
2. Open **DBSPI-Ora-15min** collector template, which contains metric 16.
3. Make five copies of the template, giving each a unique name (for example, `DBSPI-0016-SAP01, DBSPI-0016-SAP02, DBSPI-0016-SAP03, DBSPI-0016-SAP04, DBSPI-0016-SAP05`).
4. Reopen **DBSPI-Ora-15min** collector template.
5. Delete metric 16 from the **Command** text box and select the **Close** button.

6 Modify the command line of each new template using the “-i” option to specify each database by name, as shown in the table that follows:

Template	Command Line
DBSPI-00016-SAP01	dpspicao -c DBSPI-0016-SAP01 -m 16:5 -i SAP01
DBSPI-0016-SAP02	dpspicao -c DBSPI-0016-SAP02 -m 16:10 -i SAP02
DBSPI-0016-SAP03	dpspicao -c DBSPI-0016-SAP03 -m 16:10 -i SAP03
DBSPI-0016-SAP04	dpspicao -c DBSPI-0016-SAP04 -m 16:10 -i SAP04
DBSPI-0016-SAP05	dpspicao -c DBSPI-0016-SAP05 -m 16:10 -i SAP05

Changing Threshold for Microsoft SQL Server Roll-up Metrics (3026 and 3244) for Different Servers

If you have more than five servers to monitor on a system, you may want to use the tag feature described in Creating Custom Templates for Large Scale Environments on page 59. If you have fewer than five servers, you may want to set up a separate collector template for each server for a specific metric. For example, you have five servers SRV01, SRV02, SRV03, SRV04, and SRV05, for which you need different command line thresholds for different metrics. For SRV01 you would like 5; for all others you would like 10. To change the setting, you must follow these steps:

1 Select the **Message Source Templates** window.

2 Open **DBSPI-MSS7-15min** collector template, which contains metric 26.

3 Make five copies of the template, giving each a unique name (for example, `DBSPI-0026-SRV01, DBSPI-0026-SRV02, DBSPI-0026-SRV03, DBSPI-0026-SRV04, DBSPI-0026-SRV05`).

4 Reopen **DBSPI-MSS7-15min** collector template.

5 Delete metric 26 from the **Command** text box and select the **Close** button.

6 Modify the command line of each new template using the “-i” option to specify each server by name, as shown in the table that follows:

Template	Command Line
DBSPI-0026-SRV01	dbspicam -c DBSPI-0026-SRV01 -m 26:5 -i SRV01
DBSPI-0026-SRV02	dbspicam -c DBSPI-0026-SRV01 -m 26:10 -i SRV02
DBSPI-0026-SRV03	dbspicam -c DBSPI-0026-SRV01 -m 26:10 -i SRV03
DBSPI-0026-SRV04	dbspicam -c DBSPI-0026-SRV01 -m 26:10 -i SRV04
DBSPI-0026-SRV05	dbspicam -c DBSPI-0026-SRV01 -m 26:10 -i SRV05

For metric 3026, if the command line threshold is set to 5, the metric will send an alarm only if the number of active connections is more than the threshold.

For metric 3244, if the command line threshold is set to 5, the metric will log the data only if the sum of physical reads and writes is greater than or equal to the threshold value.

Creating Custom Templates for Large Scale Environments

Another advanced customization option is the `-t` (tag) option, which allows the Collector/Analyzer to use a set of templates different from the originals. This configuration provides further flexibility than would be possible in managing databases with only a few groups. Often a large number of conditions exist among the various databases across the organization.

This feature assists in organizing the various database monitoring assignments that can occur in a growing OVO environment. With hundreds of nodes managed by a number of groups, custom template groupings can effectively differentiate one group of templates from another. In such cases, you can make copies of the templates, use the tag feature to rename the group and the monitors and collector templates contained in them, then assign them to the various groups. For example, you might make an SAP group of templates and tag those templates with "SAP-" or a PeopleSoft group and tag those with "PS-."

To use this feature, make copies of the original DB-SPI templates. The names you give these new templates may contain a prefix, but they must also contain the original template name. For example, a copy of `DBSPI-0016` could be called `SAP-DBSPI-0016`. To tell the Collector/Analyzer to use this new template rather than the original template, specify the tag option on the command line.

New collector templates can also be created in this way, for example:

```
dbspicao -c SAP-DBSPI-Ora-15min -m 16 -t SAP-
```

In this case the new collector template is called "`SAP-DBSPI-Ora-15min`."

To generate automatic actions (reports) for *Oracle metrics 16, 17, and 18*, that are linked to the renamed template, you must do the following:

1. Open the monitor template conditions window; for example for Oracle metric #16, you would open DBSPI-0016.
2. In the conditions window in the Automatic Actions/Command box, insert `-t` followed by the new prefix for the existing template; the entry for the above template (SAP-DBSPI) would be:
   `dbspicao -m 16 -r 1 -i <$OPTION (dbname)> -t SAP-`

   In the command line above, the "`-t SAP-`" was added.

Turn collections Off/On According to Database Instance

The `defaults` file must be located on the managed node where you want to make the change. If this file is not on the managed node, create it. For its location on the managed node, see Data, Log, and Configuration Files on page 238.

On the managed node, edit the file to add the line:
<*first_database_instance_name*> `OFF`
<*second_database_instance_name*> `OFF`

Entry Examples

```
ora803 OFF
openview ON
SAP1 OFF
```

Instead of opening the file, you can run the command dbspicol, and the file is updated automatically. For example, you might enter:

```
dbspicol ON openview
```

dbspicol OFF SAP 1

`COLLECTION OFF` overrides any other entries in the file.

Running DB-SPI Applications from the Command Line

Each DB-SPI application function can be accessed from a command line. All applications and their parallel commands are listed below along with the platform on which each can be run.

On **UNIX** for each session in which you want to run commands from the command line, set the path to the *<OVO_commands_directory>* directory by entering the following:

```
. <OVO_commands_directory>/dbspisetpath
```

(note blank space follows period [.])

The value of <*OVO_commands_directory*> on managed nodes is as follows:

HP-UX, Linux, and Solaris DCE managed nodes:

```
/var/opt/OV/bin/OpC/cmds
```

AIX DCE managed nodes:

```
/var/lpp/OV/OpC/cmds
```

HP-UX, Linux, Solaris, and AIX HTTPS managed nodes:

```
/var/opt/OV/bin/instrumentation
```

For Windows managed nodes, at the OVO console open the Node Bank window and the Application Bank window. Select the node on which you want to run commands and drag it onto the application window to the **DB-SPI→Admin Windows→Set Path** application.

The `Set Path` application permanently adds the <*OVO_commands_directory*> path to the PATH environment variable. This extended path is set for all DOS command windows opened after running the `Set Path` application. So, the Set Path application does not need to be run more than once on each node.

Table 6 Database SPI Admin Applications

Application	Command
Enable Graphs (UNIX)	`dbspi_mw_int`
Enable Reports (UNIX)	`dbspi_mw_int -osm`
Enable Graphs (Windows)	`dbspimwi`
Enable Reports (Windows)	`dbspimwi -osm`
Disable Graphs & Reports (UNIX)	`dbspi_mwclup`
Disable Graphs & Reports (Windows)	`dbspimwc`
Check Connections	`dbspiadm conncheck -nw`

Verify Deployment	`dbspiadm verify -nw`
	or
	`dbspiverify (UNIX only)`

Display Error File	`dbspiadm dbspierror -nw`
Display Trace File	`dbspiadm trace -nw`

Database Config	`dbspicfg -e (Export)`
	`dbspicfg -i (Import)`
	`dbspicfg.sh`
	(UNIX configuration script)

Start Monitoring	`dbspicol ON [instance]`
Stop Monitoring	`dbspicol OFF [instance]`
Start Trace	`dbspicol TRACE [instance]`
Stop Trace	`dbspicol TRACEOFF [instance]`

Table 7 Oracle Applications

SQL *Plus	`dbspialo -i sqlplus`
Svr Mgr (Text)	`dbspialo -i svrmgrl`
Start Instance	`dbspialo -i dbstart`
Shutdown Instance	`dbspialo -i dbshut`
Shutdown Instance Immediate	`dbspialo -i dbshuti`
SQL Net Status	`dbspialo -i netstat`

Table 7 Oracle Applications

SQL Net Start	`dbspialo -i netstart`
SQL Net Stop	`dbspialo -i netstop`
Oracle LSNRCTL Utility	`dbspialo -i lsnrctl`

Table 8 Sybase Applications

isql	`dbspials -i isql`
Start DB Server	`dbspials -i dbstart`
Stop DB Server	`dbspials -i dbstop`
Status of Database Server	`dbspials -i dbstatus`
Sybase Tables	`dbspiasl -i tables`
Free Space	`dbspials -i freespace`

Table 9 Informix Applications

DB-Access	`dbspiali -i dbaccess`
On-Monitor	`dbspiaoi -i onmonitor`
Status of Informix Database	`dbspiali -i onstat`
On-Check	`dbspiali -i oncheck`
Query Processor	`dbspiali -i xtree`
Start DB Server	`dbspiali -i dbstart`
Stop DB Server	`dbspiali -i dbstop`
On-Perf	`dbspiali -i onperf`
Status of Database Server	`dbspiali -i dbstatus`
Find Informix Error	`dbspiali -i finderr`

Table 10 MS SQL Server Applications

Active Jobs	`dbspimjp.bat`
All Jobs	`dbspimja.bat`
NT Services	`dbspialm ntservice`
Trace	`dbspialm trace`

Table 11 Collectors*

Oracle	`dbspicao`
Informix	`dbspicai`
Sybase	`dbspicas`
MS SQL	`dbspicam`

*Please see page 51 for collector command line parameters.

Re-installing the DB-SPI Templates

When DB-SPI monitor templates are installed in OVO, they are automatically uploaded when **swinstall** is run.

To refresh any of the groups DBSPI-Core, DBSPI-Oracle, DBSPI-Informix, DBSPI-MSSQL Server, or DBSPI-Sybase, execute one or more of the following commands:

Table 12 Scripts Run by swinstall

Template Group	Scripts
Core DBSPI	`/opt/OV/bin/OpC/opccfgupld -silent* -replace -subentity /var/opt/OV/share/tmp/OpC_appl/DBSPI/core_set`
Informix	`/opt/OV/bin/OpC/opccfgupld -silent -replace -subentity /var/opt/OV/share/tmp/OpC_appl/DBSPI/inf_set`
MS SQL Server	`/opt/OV/bin/OpC/opccfgupld -silent -replace -subentity /var/opt/OV/share/tmp/OpC_appl/DBSPI/mss_set`
Sybase	`/opt/OV/bin/OpC/opccfgupld -silent -replace -subentity /var/opt/OV/share/tmp/OpC_appl/DBSPI/syb_set`
Oracle	`/opt/OV/bin/OpC/opccfgupld -silent -replace -subentity /var/opt/OV/share/tmp/OpC_appl/DBSPI/ora_set`

*Alternatively, you can use the `-verbose` option instead of the `-silent` option.

Using Filters with DB-SPI

Filters are typically used to prevent unnecessary alarms or messages. For example, if you have tablespaces which you use for read-only purposes and they are typically near 100% full, you would not want these tablespaces generating alarms. In such cases, you can use filters to prevent such unwanted alarms. You can apply filters to Oracle, MS SQL Server, Informix, or Sybase.

You can set up filters in DB-SPI for Oracle, Informix, or MS SQL Server by defining SQL `WHERE` clause fragments. DB-SPI appends the SQL `WHERE` clause fragments you define as `AND` statements to the queries that the DB-SPI collector/analyzer uses to query system tables.This

allows simple statements, `LIKE` clauses, `IN` lists, `NOT IN` lists, etc. SQL `WHERE` clause fragments can also contain a threshold value, which is especially useful for Oracle drill-down metric 217 (as shown on page 64).

To set up filters, use the **Configure DB Connections** application to open the configuration file.

1. Open the Application Bank window
2. Select **DB-SPI→Admin→Configure DB Connections**.

The tables in the DB-SPI *Oracle Reference, Informix Reference, Sybase Reference,* and *MS SQL Reference* guides provide a comprehensive list of metric specifications.

> The user must **not** specify a table name in the filter specification except for Informix metrics 1061 and 1261.

Syntax Checking

The **Configure DB Connections** application checks the syntax of the filter each time the configuration file is saved.

Syntax for Oracle

The syntax for Oracle filtering using Configure DB Connections is:

```
ORACLE
  HOME "<ORACLE_HOME>"
    DATABASE <name> CONNECT "<user/password>"
      FILTER <metic_number> "<SQL where qualifier>"
    ...
    DATABASE <name> CONNECT "<user/password>"
      FILTER <metic_number> "<SQL where qualifier>"
    ...
```

Example

```
SYNTAX_VERSION 4

ORACLE
  HOME "/opt/oracle/9.2.0"
    DATABASE sap CONNECT "system/manager"
    FILTER 4   " Username NOT IN ('Guy','Barry')"
    FILTER 5   " Owner NOT IN('Guy','Debbie','OPC_OP')"
    FILTER 6   " Tablespace_name NOT LIKE 'SAP1001%' and
      tablespace_name NOT BETWEEN 'TS001' and 'TS010'"
    FILTER 16 " Owner <> 'WAREHOUSE' and Segment_name NOT LIKE
      'SAP%'"
    FILTER 42.1 "TABLE_NAME NOT LIKE 'SAP%"
    FILTER 42.3 "INDEX_NAME NOT LIKE 'SAP%'"
    FILTER 217 "((EXTENTS/MAX_EXTENTS)*100 > 80) and
      MAX_EXTENTS <> 0"
    FILTER 67 " SEGMENT_NAME <> 'R0'"
    FILTER 206 "Tablespace_name NOT LIKE 'SAP1005%' and
      tablespace_name NOT BETWEEN 'TS010' and 'TS100'"
```

Table 13 Filter Design for Oracle

Metric	Area	Object	Table	Columns	Sample WHERE Clause Fragment
3	SPACE MANAGE-MENT-Misc.	Table Space Names	dba_tablespace	tablespace_name	Tablespace_name ='system'
203	SPACE MANAGE-MENT-Misc.	Table Space Names	dba_tablespace	tablespace_name	Tablespace_name ='system'
4	SPACE MANAGE-MENT-Misc.	User Names	dba_users	username	Username NOT IN ('Guy','Debbie' , 'Barry')
5	SPACE MANAGE-MENT-Misc.	Owners	dba_segments	owner	Owner NOT IN ('Guy','Debbie' , Barry')
6	SPACE MANAGEMENT-TblSpaces	Table Space Names	dba_data_files dba_free_space	tablespace_name	Tablespace_ name NOT LIKE 'SAP1001%' and tablespace_name NOT BETWEEN 'TS001' and 'TS0101'
7		Table Space Names	dba_tablespaces	tablespace_name	Same as 6
8		Table Space Names	dba_data_files	tablespace_name	Same as 6
9	SPACE MANAGEMENT	Table Space Names	dba_tablespace	tablespace_name	Tablespace_name <>'SAP'
11		Table Space Names	dba_free_space	tablespace_name	Same as 6
16	SPACE MANAGEMENT-Segments	Table Space	dba_tablespace	tablespace_name	tablespace_name like 'SAP%'
17		Table Space, Object_ Owner, Object_Name	dba_segment	tablespace_name owner segment_name segment_type extents max_extents	Owner<> 'WAREHOUSE' and Segment_ name like 'SAP%'
18		Table Space, Object_ Owner, Object_Name	dba_segment	tablespace_name owner segment_name	Owner<> 'WAREHOUSE' and Segment_ name like 'SAP%'
42.1	Tables	Table Name	dba_tables	Any column in dba_tables	TABLE_NAME NOT LIKE 'SAP%'

Table 13 Filter Design for Oracle

Metric	Area	Object	Table	Columns	Sample WHERE Clause Fragment
42.3	Indexes	Index Name	dba_indexes	Any column in dba_indexes	INDEX_NAME NOT LIKE 'SAP%'
67	ROLLBACK SEGMENTS	Rollback Segment Names	dba_rollbacks_segs	segment_name	Segment_name <> 'R0'
80	Constraints	Constraint Names	dba_contraints	contraint_name	CONSTRAINT_NAME NOT LIKE 'SYS_C00%'
100	SQL Query Monitoring	User Names	dba_users	username	username NOT IN ('SYSTEM', 'SYS')
NOTE: The filter for metric 100 affects Oracle metrics 101 to 107 as metric 100 acts as the data gatherer for collection of these metrics.					
206	SPACE MANAGEMENT-TblSpaces	Table Space Names	dba_data_files dba_free_space	tablespace_name	Tablespace_name NOT LIKE 'SAP1005%' and tablespace_name NOT BETWEEN 'TS010' and 'TS0100'
216	SPACE MANAGEMENT-Segments	Table Space	dba_tablespace	tablespace_name	tablespace_name like 'SAP%'
217		Table Space, Object_Owner, Object_Name	dba_segment	tablespace_name owner segment_name segment_type extents max_extents	Owner<> 'WAREHOUSE' and Segment_name like 'SAP%'
218		Table Space, Object_Owner, Object_Name	dba_segment	tablespace_name owner segment_name	Owner<> 'WAREHOUSE' and Segment_name like 'SAP%'

Syntax for MS SQL Server

```
SYNTAX_VERSION 4

MSSQL
SERVER "SAPSVR1"
    Server mssql "system/manager"
    FILTER 209  "instance_name not in ('master')"
    FILTER 215  "name not like 'MSDB%'"
    FILTER 216  "SetOptName not in ('master','"
```

```
FILTER 266 "instance_name not in ('master')"
```

Table 14 Filter Design for MS SQL Server

Metric	Object	Table	Columns	Sample WHERE Clause Fragment
3209	Database Name	master..sysperfinfo	instance_name	instance_name not in ('master','pubs')
3215	Virtual Device Name	sysdevices	sysdevices.name	name not like 'MSDB%'
3216	Database Name	master..sysperfinfo	name	instance_name not in ('master', 'pubs')
3218	Database Name	master..sysdatabases	name	name not in ('master', 'model')
3227	Login Names	master..sysprocesses	loginame = 'sa'	name not in ('master', 'model')
3233	Database Name	master..sysdatabases	name	name not in ('master','model')
3234	Database Name	master..sysdatabases	name	name not in ('master','model')
3230	Database Name	master..sysdatabases	name	name not in ('master', 'model', 'pubs')
3264	Database Name	master..sysperfinfo	instance_name	instance_name not in ('master')
3266	Database Name	master..sysperfinfo	instance_name	instance_name not in ('master')
3267	Database Name	master..sysperfinfo	instance_name	instance_name not in ('master')
3270	Object Type (Extent, Key, Page, Table, RID, Database)	master..sysperfinfo	instance_name	instance_name not in ('master')
3271	Object Type (Extent, Key, Page, Table, RID, Database)	master..sysperfinfo	instance_name	instance_name != 'RID'

Table 14 Filter Design for MS SQL Server

Metric	Object	Table	Columns	Sample WHERE Clause Fragment
3272	Object Type (Extent, Key, Page, Table, RID, Database)	master..sysperfinfo	instance_name	instance_name != 'RID
3273	Object Type (Extent, Key, Page, Table, RID, Database)	master..sysperfinfo	instance_name	instance_name != 'RID
3277	Job Name	msdb..sysjobs	sysjobs.name	name <> 'Test'

Syntax for Informix

The syntax for Informix filtering using Configure DB Connections is:

```
SYNTAX_VERSION 5

INFORMIX

  INFORMIX
    HOME "/opt/informix/9.2"
    SERVER "informix920"
      ONCONFIG "onconfig.informix920"
      CONNECT "informix/dbspi2004"
      SQLHOSTS "/opt/informix/9.20/etc/sqlhosts"
      FILTER 7 "username NOT IN ('guyr','fred')"
```

The following table shows the metrics, the objects to filter on, the actual tables and columns that will be filtered and a sample WHERE clause fragment.

The user should NOT specify a table name in the filter specification except for Informix metrics 1061 and 1261.

Table 15 Filter Design for Informix

Metric	**Area**	**Object**	**Table**	**Columns**	**Sample WHERE Clause Fragment**
1007	Status	User Name	syssessions	username	username NOT IN ('Guy','Debbie', 'Barry')
1017	Space Management	Database Space Name	sysdbspace	name	name NOT IN ('SAP01','SAP02')
1217	Space Management	Database Space Name		name	name NOT IN ('SAP01','SAP02')
1020	Space Management	Chunk Number	syschunks	chknum	chknum NOT IN (23,56,78)

Table 15 Filter Design for Informix

Metric	Area	Object	Table	Columns	Sample WHERE Clause Fragment
1022	Space Management	Database Space Name Owner Table Name	systabnames sysptnext	dbsname owner tabname	owner <> 'SAP' and tabname NOT LIKE 'SAP01*'
1061	Transactions	User Name Session ID	syssessions syssesprof	username sid	username NOT IN ('Guy','Debbie', 'Barry') AND syssesprof.sid 200
1261	Transactions	User Name Session ID	syssessions syssesprof	username sid	username NOT IN ('Guy','Debbie', 'Barry') AND syssesprof.sid 200
1076	Locking	User Name	syssessions	username	username NOT IN ('Guy','Debbie', 'Barry')

The sid field for Informix metrics 1061 and 1261 must be fully qualified with the tablespace name `syssesprof`.

Syntax for Sybase

```
SYNTAX_VERSION 4
SYBASE
SERVER "SAPSVR1"
    Server sybase "system/manager"
    FILTER 2206 “name <> ‘test1’”
```

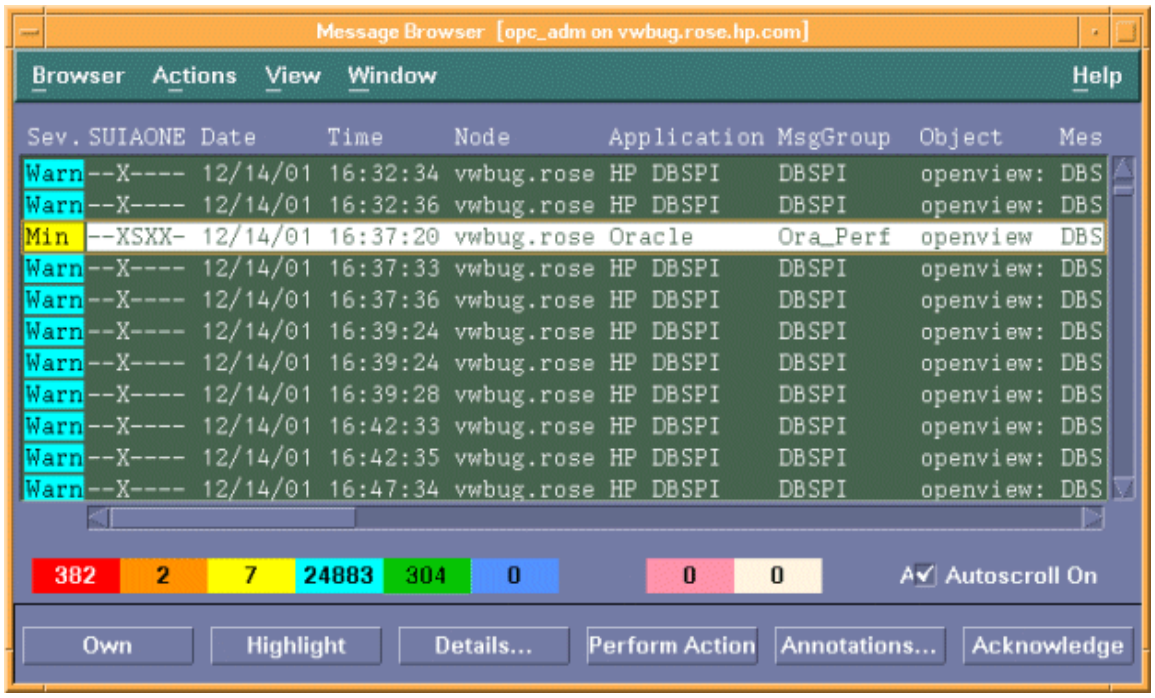
Table 16 Filter Design for Sybase

Metric	Object	Table	Columns	Sample WHERE Clause Fragment
2005	Database Name	master..sysdatabases	name	name not like 'm%'
2205	Database Name	master..sysdatabases	name	name not like 'm%'
2206	Database Name	master..sysdatabases	name, dbid	name <> ‘test1’

Some templates also have actions defined with threshold violations or error conditions that automatically cause reports to be generated. You can find such a report in the Annotations area of the Message Details. You can also manually generate reports by using the DBSPI Reports application. These two types of reports are:

- Automatic Action Reports
- Application Bank Reports

Figure 18 Checking for indications of automatic action reports in the OVO Message Browser



Automatic Action Reports

Many metrics generate Automatic Action Reports. These reports are generated as soon as an alarm is triggered in OVO. The collector/analyzer either executes the report directly or saves information in `/var/opt/OV/dbspi/history`. This file is updated every time a metric is analyzed.

How you know a report has been generated

When an Automatic Action Report is executed from OVO, the file is read and sent as a report or the report is executed directly. These reports are indicated by an “X” in the “A” cell of the OVO Message Browser column labeled SUIAONE (see illustration on the previous page). An “S” under the “A” flag indicates that the report was generated successfully and is waiting in the Annotations area of the Message Details.

How to view the report

To view the report, simply click on Annotations in the OVO Message Browser. The Instruction Text for the metric explains the report. Column descriptions provide further clarification.

How Automatically Generated and Manually Generated Reports Differ

Automatic Action Reports show the state of the system moments after the alarm occurred. You can manually generate a report that shows the current state of the system, by dragging the managed node from the Node Bank onto the desired report in the Application Bank (see the next section for details on Application Bank reports).

Application Bank Reports

Application Bank Reports run for each database configured on the system, in contrast to Automatic Action Reports which are generated for a single database instance. These reports, generated from an application group, reflect the current state of the system.

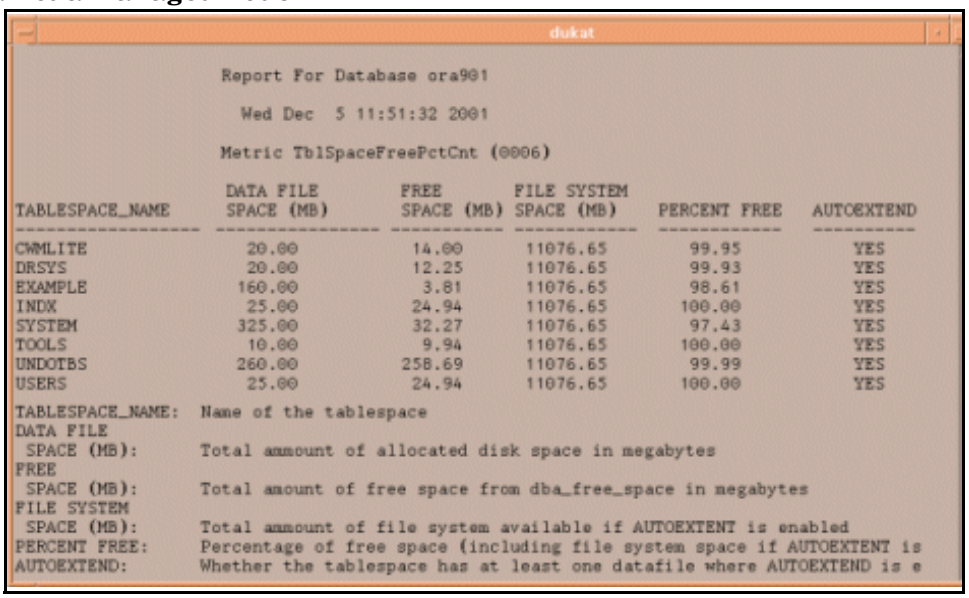
To generate a report in OVO, simply drag the managed node from the Node Bank window for which you want the report on to the desired report in the Application Bank window.

Figure 19 Oracle application bank reports show reports on all database instances on a selected managed node

Sample Application-Generated Report

The following example shows the format of a DB-SPI application-generated report. The same report, only for a specific database instance is generated automatically, in the case of an alarm condition.

Figure 20 Report generated by running an Oracle Report Application (Metric 0006) against a managed node



dukat

Report For Database ora901

Wed Dec 5 11:51:32 2001

Metric TblSpaceFreePctCnt (0006)

TABLESPACE_NAME	DATA FILE SPACE (MB)	FREE SPACE (MB)	FILE SYSTEM SPACE (MB)	PERCENT FREE	AUTOEXTEND
CWMLITE	20.00	14.00	11076.65	99.95	YES
DRSYS	20.00	12.25	11076.65	99.93	YES
EXAMPLE	160.00	3.81	11076.65	98.61	YES
INDX	25.00	24.94	11076.65	100.00	YES
SYSTEM	325.00	32.27	11076.65	97.43	YES
TOOLS	10.00	9.94	11076.65	100.00	YES
UNDOTBS	260.00	258.69	11076.65	99.99	YES
USERS	25.00	24.94	11076.65	100.00	YES

TABLESPACE_NAME: Name of the tablespace
DATA FILE
SPACE (MB): Total ammount of allocated disk space in megabytes
FREE
SPACE (MB): Total amount of free space from dba_free_space in megabytes
FILE SYSTEM
SPACE (MB): Total ammount of file system available if AUTOEXTENT is enabled
PERCENT FREE: Percentage of free space (including file system space if AUTOEXTENT is
AUTOEXTEND: Whether the tablespace has at least one datafile where AUTOEXTEND is e

Report-generating applications are as follows:

- DBSPI Informix Rep
- DBSPI Oracle Rep (HP-UX, Linux, AIX & Solaris)
- NT-DBSPI Oracle Rep (Windows)
- DBSPI MSS6 Rep (MS SQL 6.5)
- DBSPI MSS7 Rep (MS SQL 7.x)
- DBSPI Sybase Rep

Investigating Performance Problems Using OpenView Performance Agent

If you use OpenView Performance Agent, database metrics can help you monitor specific areas that affect system performance. OpenView Performance Agent's alarm generating capabilities are well suited to narrow in on system performance problems. Observing the alarm-generating conditions occurring with specific metrics over time, you can discover what factors, including those not directly database-caused but more system-related, are contributing most to the problem.

The following are hypothetical examples of how an OpenView Performance Agent alarm definition file might be used to add to database performance monitoring.

Excessive Table Scans

A table scan occurs when the database does not use an index to access a row or group of rows. Because table scans slow system performance, this is a problem in many database environments.

Even though most sites make a substantial effort to avoid table scans, they may still occur fairly often. By using both OpenView Performance Agent system metrics and the Database SPI metrics, you can often identify table scans:

- OpenView Performance Agent system metrics that might affect table scans are CPU utilization and peak disk utilization.
- DBSPI metric FulLngTblScnRate_030 in an OpenView Performance Agent `alarmdef` file more precisely pinpoints the problem.

Monitoring Database Table Scans

The following example shows how these metrics could be used to define an OpenView Performance Agent alarm:

```
############ alarm definitions
USE "ORADB_KIMBALL_ORA733"
ALARM (ORADB_METRICS:E030_FulLgTblScnRate >= 80 AND
       SCOPE:GLOBAL:GBL_DISK_UTIL_PEAK >= 90 AND
       SCOPE:GLOBAL:GBL_CPU_TOTAL_UTIL >= 90) FOR 5 MINUTES
TYPE="ORACLE"
  START {
     YELLOW ALERT "Table scan rate too high for 'ora733' (E030_FulLgTblScnRate)"
  }
  REPEAT EVERY 30 MINUTES {
     YELLOW ALERT "Table scan rate too high for 'ora733' (E030_FulLgTblScnRate)"
  }
END RESET ALERT "End alarm - Table scan rate for 'ora733'"
############ End alarm definitions

This example is for a database named ora733. The USE statement is for purposes of
specifying a data source other than the SCOPE data source. To determine the data sources
defined on your system, execute the command mwa status and look for the list of data
sources as shown in the following example:
PIDDATA SOURCE
Running rep_server24354 ORADB_KIMBALL_ORA733
Running rep_server24351SCOPE
```

The `SCOPE` metrics must be a fully qualified metric name :`SCOPE`:<*class_name*>:<*metric_name*>

Monitoring Database Buffer Activity

Database buffers are critical to performance. If an insufficient number are configured, the database must perform a large number of physical reads and writes. User SQL requests can become blocked, waiting on buffers, if too few are configured.

In addition, kernel memory also affects database performance. Within kernel memory, several different object types reside, such as the kernel buffer cache, Oracle Shared Memory Area (SGA), application and system processes. Allocating sufficient memory for each object type ensures acceptable performance, while an insufficient allocation could cause excessive physical reads and writes, degrading performance. Too much memory allocation to any object type, also causes problems, forcing other objects out (to page out) to the swap area, again affecting performance.

The following example shows both critical system resource and database buffer activity monitoring.

```
############ alarm definitions
USE "ORADB_KIMBALL_ORA733"
```

```
ALARM (ORADB_METRICS:E023_CurBufCacHitPct <= 80 AND
       SCOPE:GLOBAL:GBL_MEM_UTIL <= 40 AND
       SCOPE:GLOBAL:GBL_MEM_PAGEOUT_RATE >= 80) FOR 5 MINUTES TYPE="ORACLE"
  START {
     RED ALERT "DB Block buffers contention for'ora733'(E023_CurBufCacHitPct)"
  }
  REPEAT EVERY 30 MINUTES {
     RED ALERT "DB Block buffers contention for'ora733'(E023_CurBufCacHitPct)"
  }
  END RESET ALERT "End alarm - DB block buffers contention for'ora733'"
############ End alarm definitions
```

Consolidated Monitoring

Using the above examples, both alarm definitions could be consolidated into an OpenView Performance `alarmdef file`. For example, you could create a file:

`/var/opt/OV/dbspi/dsi/oracle/ora733.alarm` that contained the two examples:

```
############ alarm definitions
USE "ORADB_KIMBALL_ORA733"
ALARM (ORADB_METRICS:E030_FulLgTblScnRate >= 80 AND
       SCOPE:GLOBAL:GBL_DISK_UTIL_PEAK >= 90 AND
       SCOPE:GLOBAL:GBL_CPU_TOTAL_UTIL >= 90) FOR 5 MINUTES TYPE="ORACLE"
  START {
     YELLOW ALERT "Table scan rate to high for 'ora733' (E030_FulLgTblScnRate)"
  }
  REPEAT EVERY 30 MINUTES {
     YELLOW ALERT "Table scan rate to high for 'ora733' (E030_FulLgTblScnRate)"
  }
END RESET ALERT "End alarm - Table scan rate for 'ora733'"


ALARM (ORADB_METRICS:E023_CurBufCacHitPct <= 80 AND
       SCOPE:GLOBAL:GBL_MEM_UTIL <= 40 AND
       SCOPE:GLOBAL:GBL_MEM_PAGEOUT_RATE >= 80) FOR 5 MINUTES TYPE="ORACLE"
  START {
     RED ALERT "DB Block buffers contention for'ora733'(E023_CurBufCacHitPct)"
  }
  REPEAT EVERY 30 MINUTES {
     RED ALERT "DB Block buffers contention for'ora733'(E023_CurBufCacHitPct)"
  }
  END RESET ALERT "End alarm - DB block buffers contention for'ora733'"
############ End alarm definitions
```

You can add the file to the OpenView Performance Agent `alarmdef (/var/opt/perf/alarmdef)` by using an include statement at the end of the file.

```
include /var/opt/OV/dbspi/dsi/oracle/ora733.alarm
```

After adding the alarm definition restart the OpenView Performance Agent alarm agent using the command

```
mwa restart servers
```

A complete description of the OpenView Performance Agent alarm syntax can be found in the *HP OpenView Performance Agent User's Manual*.

Check the Installed Database SPI Nodes with License Count

You can use an OVO reporting utility to check the number of templates you have installed on your managed nodes. In reviewing the number of templates per managed node, you can see if you have consistently installed templates across your managed systems. In addition, by running this report, you can also ensure that the number of licenses you have purchased is in compliance with the report results.

To run the report:

1. In the OVO Node Bank window select the node or node group that you want to check.
2. From the Actions menu select **Utilities→Reports...**.
3. In the Reports window among the reports listed select **DBSPI License Check**, the method for the output display, and click **OK**.

Changing from one OpenView Agent version to Another

DB-SPI supports the new HTTPS agent, which comes with out-of-the-box HTTP authentication and SSL encryption with server and client certificates, guaranteeing secure communication with HP OpenView Operations central management servers. The new agent is available for all popular operating system platforms, including HP-UX, Microsoft Windows, Linux, Sun Solaris, IBM AIX, and Tru64 UNIX.

Updates: When you update from OVO 7.x to OVO 8.x and install the new agent, no manual configuration is necessary.

Reconfiguration: If you change the OVO 8.0 agent either from DCE to HTTPS or from HTTPS to DCE, complete the following steps in preparation for the change:

1. Open the Application Bank and select the DBSPI application group.
2. Double-click **DBSPI-Admin**.
3. Drag and drop the managed node or node group for which you want the change to occur onto the Migrate Agent application.

Changing from OpenView Performance Agent to the OpenView Subagent

The HP OpenView Database SPI can detect whether or not you are using OpenView Performance Agent (also known as MeasureWare Agent). If you are, your new DB-SPI installation will automatically use it as well. As a result, if you use PerfView, your new installation also supports that configuration.

If for any reason you decide that you want to use the new HP OpenView subagent included with OVO 8.0, you can configure managed nodes to do so. Note that this configuration does not support PerfView.

To override the use of OpenView Performance Agent, set up an empty file named nocoda.opt and store it on the managed node in a specific location. The location will vary according to the managed node operating system as shown below.

Table 17 Location for File Overriding Openview Performance Agent

Managed Node O/S	File Location
HP-UX, Linux, and Solaris	`/var/opt/OV/conf/dsi2ddf/nocoda.opt`
AIX	`/var/lpp/OV/conf/dsi2ddf/nocoda.opt` (DCE Agent) `/var/opt/OV/conf/dsi2ddf/nocoda.opt` (HTTPS Agent)
Windows	`\usr\ov\conf\dsi2ddf\nocoda.opt`

To create the file:

1. (If necessary) on the managed node according to the path shown in the preceding table, create the `dsi2ddf` directory.
2. Use a text editor to open a new file.
3. Save the file as `nocoda.opt` in the managed node's `dis2ddf` directory.

Overriding OVO Thresholds by Configuring Local System Thresholds

An option for overriding OVO thresholds allows you to define local thresholds, which bypass the settings contained in OVO monitor templates deployed on the node. Using this option, you can configure the Database SPI to generate fewer alarms and, in turn, prevent generation of unwanted messages from a specific system and/or database.

You establish local override settings by creating a simple ASCII text configuration file, which you save as `override.cfg`. Within this file you can define a specific threshold for any metric collected on the local system. In the case of a *maximum* type threshold, you would establish a local threshold *higher* than that set within the OVO monitor template. Conversely, a local *minimum* threshold would be *lower* than that defined in the OVO minimum threshold monitor template.

This option also allows a local database administrator to define every metric threshold (see Overriding All Thresholds on page 78) for monitoring all databases running on the system for which he/she is responsible. Metric thresholds are easily reviewed and adjusted as necessary since they are all contained in a single file.

When you finish with the file, save it *on the managed node* in the default directory, according to the node's operating system as follows:

Table 18 Overriding Template Thresholds with Individual Files

HP-UX, Linux, Solaris, Tru64	`/var/opt/OV/dbspi/`
AIX	`/var/lpp/OV/dbspi/` (DCE Agent) `/var/opt/OV/dbspi/` (HTTPS Agent)
Windows	`\usr\OV\dbspi\`

Please refer to the sections that follow for syntax and an example.

File Syntax

File syntax follows the conventions:

- Lines beginning with the pound sign (#) are recognized as comments.
- Blank lines are ignored.
- White space is used as a delimiter, but is otherwise ignored; for example, carriage returns and extra spaces are unrecognized except as token separators.

```
#  File-syntax = Zero or more <monitor_statement>
#                <End-of-File>
#
#  <monitor statement> =  MONITOR "<monitor name>"
#                         <monitor type>
#                         MSGCONDITIONS
#                         one or more <condition statement>
#
#  <monitor type> = { MAXTHRESHOLD | MINTHRESHOLD }
#
#  <condition statement> = CONDITION
#                           [ OBJECT "<object name>"]
#                          THRESHOLD <threshold value>
#
# <monitor name> = The name of the monitor
#                  expecting the value --
#                  including the TAG
#                  (case insensitive)
#
# <object name> = (optional) The name of the object
#                 generated by DB-SPI.
#                 (case insensitive)
#                 If OBJECT is omitted, the THRESHOLD
#                 applies to the entire monitor.
#                 If OBJECT is included, the CONDITION
#                 must match the MONITOR and OBJECT.
#
# <threshold value> = a numeric (double)
#
```

Overriding Specific Thresholds

In the example `override.cfg` file below, a threshold is re-defined locally for two Oracle metrics: Metric 0217, Segment Maximum Extent Count; and Metric 0022, Total Buffer Cache Hit percentage.

The first metric has a maximum threshold setting of 80(%) in the monitor template. This threshold is re-set locally with two conditions, increasing one to 85(%), and the other to 90(%). For the second metric, the minimum threshold is re-set locally, decreasing it from the monitor template setting of 90(%) to 80(%).

To create the file:

1. Using a text editor, open an empty file.
2. Insert entries as shown below for each monitor template deployed to the managed node.
3. Save the file as `override.cfg` in the appropriate default directory (see page page 76).

File Entry Examples

```
MONITOR "DBSPI-0217"
       MAXTHRESHOLD
       MSGCONDITIONS
                CONDITION
                     OBJECT "ora217:segmentA"
                     THRESHOLD 85
                CONDITION
                     OBJECT "ora217:segmentB"
                     THRESHOLD 90
MONITOR "DBSPI-0022"
       MINTHRESHOLD
       MSGCONDITIONS
                CONDITION
                        OBJECT "oradb"
                        THRESHOLD 80
```

Overriding All Thresholds

If you are the database administrator for a specific system and want full control over every threshold setting on the system, you can set up a file that is easy to modify whenever necessary. The method described below addresses this scenario by suggesting that you virtually clear out every monitor template threshold before you start.

Local thresholds (in the `override.cfg` file) work as primary thresholds; the monitor template thresholds are secondary. If a value exceeds the local threshold, it is passed through to the secondary threshold (of the template), which then generates the message/alert.

The simplest way to establish local thresholds for all metrics is to first re-set all monitor templates to *opposite extreme values*. By so doing, you will never find it necessary to open a template to ensure that your locally set threshold is higher or lower than the maximum or minimum setting of the template whose threshold you are overriding.

The steps for establishing local thresholds for *all collected metrics* on a system are as follows:

1. Open each OVO monitor template you are deploying to the node and re-set the threshold to an opposite extreme value; for example:
   a maximum threshold could be set at the low value of 0.5,
   a minimum threshold could be set at the high value of 9999999.
2. Using your text editor, open an empty file and in which you create an entry for every metric collected on the system.
3. Save the file as `override.cfg` in the appropriate directory (see page 76) on the managed node.

Modifying Message Text to Match Locally defined Thresholds

For a metric whose alarm generates a message showing the threshold value, you can modify the original message text to reflect a local, modified threshold. For this purpose you add an option to the message text definition that specifies a local or database instance-specific threshold be shown whenever one is encountered (as in the example below). The syntax for overriding the default threshold is:

```
<$THRESHOLD $OPTION (local_threshold)
```

For example, Oracle metric 203 has message text defined as follows:

```
DBSPI-0203.1: Free extents (<$VALUE>) for tablespace
<$OPTION(tablespace_name)> too low (\<=<$THRESHOLD>) for
<$OPTION(dbname)>.
```

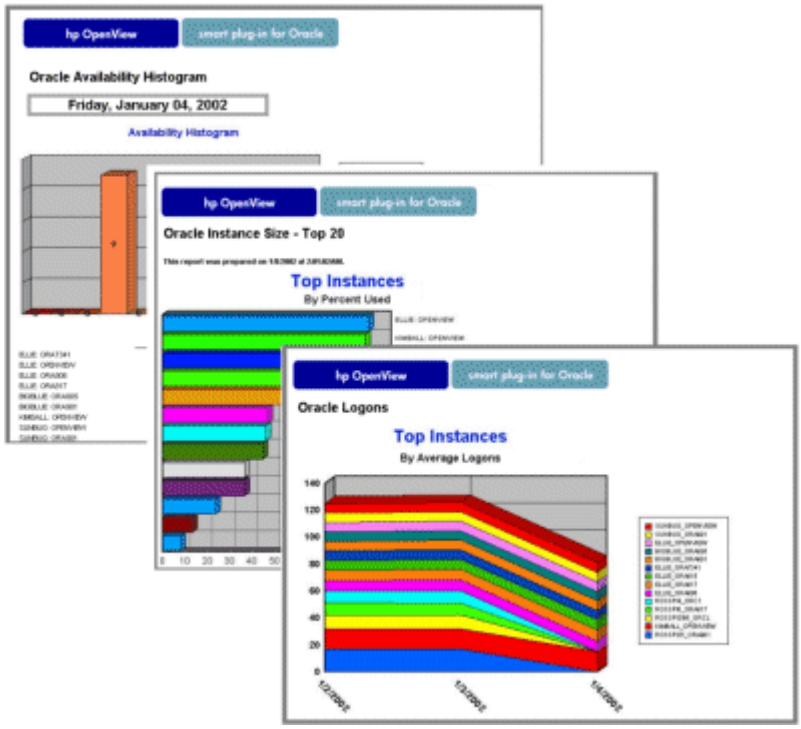
You would change the message text for the template to:

```
DBSPI-0203.1: Free extents (<$VALUE>) for tablespace
<$OPTION(tablespace_name)> too low (\<= OVO threshold <$THRESHOLD> and
local threshold <$OPTION(local_threshold)>) for <$OPTION(dbname)>.
```

4 Using the Database SPI with OpenView Reporting and Graphing Products

This chapter covers the Database SPI *data collections* that allow *reporting* and *graphing*. Report- and graph-generating metrics are contained in Quick Start in the sub-group labeled *Reporter*. Consequently, when you deploy Quick Start, you also deploy policies that enable OVO to generate reports, viewable the following day, and graphs viewable as operator actions and manually generated.

Figure 21 Database SPI reports generated by Reporter can vividly show, among other things, availability, instance size, and log-on trends to help you formulate an overall picture of database activity



Reporting and Graphing Data: How it is Stored and Accessed

The current version of the Database SPI collects and stores reporting and graphing data in a different format. This new format, unlike the earlier formats, does not rely on the managed node system name to uniquely identify the datasource. For example, in the past Oracle graphing metrics collected from system Kimball were stored in datasource DBSPI_ORA_KIMBALL_GRAPH. Now those Oracle graphing metrics are stored in datasource DBSPI_ORA_GRAPH. Note, however, that the Database SPI continues to collect and store data in earlier formats that are compatible with earlier products, such as Reporter.

The advantage to storing the collected reporting and graphing data in generic datasources is that template creation becomes streamlined. Now you can set up a generic template to be used over and over again because its datasource does not include a system name. Each generic datasources includes a column reserved for the database instance name. This column's entry takes the place of the system name as the unique identifier of the collected data's source.

The collected metrics are largely unchanged (they are specifically identified after the Table "Datasource Names for Graphing Metrics" on page 82); but customizing their collection has become easier. Any template you create for specific metric collections can be used over and over again for different database instances on different systems.

The generic datasource names are according to the four database types: Informix, MS SQL Server, Oracle, or Sybase. In addition to database type, the datasources are categorized by the type of metrics collected. Metrics fall into three groups: reporting, graphing, and user-defined [metrics] (UDMs) as shown in the tables that follow.

Table 19 Database Types & Generic Datasource Names

Database Type	Datasource Names for Reporting Metrics
Informix	DBSPI_INF_REPORT
Microsoft SQL Server	DBSPI_MSS_REPORT
Oracle	DBSPI_ORA_REPORT
Sybase	DBSPI_SYB_REPORT

Database Type	Datasource Names for Graphing Metrics
Informix	DBSPI_INF_GRAPH
Microsoft SQL Server	DBSPI_MSS_GRAPH
Oracle	DBSPI_ORA_GRAPH
Sybase	DBSPI_SYB_GRAPH

Database Type	Datasource Names for User Defined Metrics (UDMs)
Informix	DBSPI_INF_UDM
Microsoft SQL Server	DBSPI_MSS_UDM
Oracle	DBSPI_ORA_UDM
Sybase	DBSPI_SYB_UDM

Generic Datasource Graphing Metrics

Generic datasource reporting metrics are not included in this section because they are inaccessible. Graphing metrics, however, are accessible for selection for each graph.

The data stored in the Database SPI generic datasources are listed in the following sections so that you are aware of those you can choose. Some graphing/reporting metrics are not included in the new generic datasources. For metric definitions for Oracle, Microsoft SQL Server, Informix, and Sybase, refer to the respective *Reference Guides*.

Informix Graphing Metrics for Use with DBSPI_INF_GRAPH:

INSTANCENAME, I001_PROCESSSTATUS, I002_INSTANCESTATUS, I003_ACCESSIBLE, I005_USERLOGONCNT, I007_NUMSESSIONS, I015_PLOGPGFRPCT, I031_SHMTOT, I032_SHMVIRTSEGS, I033_SHMUSEDPCT, I036_SHMDATABUFMDPCT, I037_TOTREADCACHTPCT, I038_TOTWRITCACHTPCT, I039_CURREADCACHTPCT, I040_CURWRITCACHTPCT, I042_FGWRITES, I043_LRUWRITES, I044_CHUNKWRITES, I047_SEQSCANSRATE, I050_SORTDISKRATE, I051_SORTTOTALRATE, I052_TOTSORTMEMRYPCT, I053_CURSORTMEMRYPCT, I054_SORTMAXSPACE, I056_TXNTHROUGHPUT, I057_COMMITRATE, I058_ROLLBACKRATE, I059_TOTROLLBACKPCT, I060_CURROLLBACKPCT, I064_LLOGFR, I065_LLOGFRPCT, I066_LLOGNONBKUPPCT, I069_CKPTRATE, I070_CKPTDURATION, I071_CKPTWAITSRATE, I073_LOCKAVAILPCT, I074_LOCKWTSREQPCT, I075_DEADLOCKS, I076_SESSIONBLKLOCK

Microsoft SQL Server Graphing Metrics for Use with DBSPI_MSS_GRAPH:

INSTANCENAME, M001_CACHEHITPCT, M002_CACHEFREEBUFPCT, M003_LOGLOGICALIOAVG, M004_BATCHWRITESRATE, M005_PAGEREADSRATE, M006_SNGLPGEWRTSRATE, M007_READSOUTSTDRATE, M008_WRITSOUTSTDRATE, M009_TRANSACTIONRATE, M010_RAMGREFFCTVPCT, M011_USERCONNECTPCT, M012_RASLOTSUSEDPCT, M013_LOCKSINUSEPCT, M014_BLCKDPROCESSCNT, M017_CMDQUEUELENPCT, M019_PROCCACHEUSDPCT, M020_PROCCACHEACTPCT, M021_SINGLPGEWRITPCT, M022_BUFCHAINAVGLEN, M023_READWRITEERRCNT, M024_PACKERERRORCNT, M025_CPUUSEDPCT, M026_ACTIVECONNTNPCT, M028_SUSPECTDBCNT, M029_OPENDBPCT, M031_NUMUSERSCNT, M051_FULLSCANSRATE, M052_INDXSEARCHSRATE, M053_PGESALLOCTDRATE, M054_EXTNTSALLOCRATE, M055_PAGESPLITSRATES, M056_TBLLCKESCALRATE, M064_DBACTIVTRANSCNT, M066_DBLOGGROWTHSCNT, M067_DBLOGSHRINKSCNT, M068_LATCHWAITSRATE, M069_AVGLATCHWAITTIM, M070_LOCKTIMEOUTRATE, M071_DEADLOCKSRATE, M072_LOCKSWAITRATE, M073_LOCKAVGWAITTIME, M074_BATCHREQSTSRATE, M075_LOCKMEMORYPCT, M076_CURAVGWAITLATCH

Oracle Graphing Metrics for Use with DBSPI_ORA_GRAPH:

INSTANCENAME, E001_DBINSTANCESTAT, E004_USERSTMPDFLTCNT, E005_OBJCTSFORIGNCNT, E006_TBLSPFREEPCTCNT, E007_TBLSPCSTATUSCNT, E008_TSBREADRATIOCNT, E009_TSTMPEXNTPCTCNT, E011_TBLSPCFRGMNTCNT, E016_SEGMNTEXTENDCNT, E017_SEGMAXEXTENTCNT, E018_SEGEXTRAPIDCNT, E019_SORTDISKRATE, E020_SORTMEMORYPCT, E021_BUFFERBUSYPCT, E022_TOTBUFCACHITPCT, E023_CURBUFCACHITPCT, E024_EQWAITSREQPCT, E026_DICTCACHEHITPCT, E027_LIBCACHRELODPCT, E028_LOCKSUSEDPCT, E029_SESSWAITLCKCNT, E030_FULLGTBLSCNRATE, E031_OPENCRSRPCTCNT, E032_REDOLGSPCREQCNT, E033_REDOALOCLTCHPCT, E034_REDOCOPYLTCHPCT, E035_BCKGNDCKPTRATE, E037_USERLOGONCNT, E038_LTCHOVRLIMITCNT,

E039_LIBCACGETHITPCT, E040_LIBCACPINHITPCT, E041_FULSHTBLSCNRATE, E042_UNLYZTBLINDXPCT, E043_EQTIMEOUTREQPCT, E044_COMMITRATE, E045_SHRDPOOLFREEPCT, E046_ROWFETCBYIDXPCT, E047_TABLESCACHEDCNT, E048_CHANDROWFTCHPCT, E049_USERCALLRATE, E050_RCSVUSRCALRATIO, E051_SORTROWSAVGCNT, E052_SORTTOTALRATE, E054_ROLLBACKRATE, E056_ARCHVFREESPCCNT, E057_ARCHIVEFREQRATE, E058_ARCHVFREESPCPCT, E059_CURSORCACHEPCT, E062_BKGRDUMPSPCEPCT, E064_USERDUMPSPACPCT, E065_COREDUMPSPACPCT, E066_ALERTLOGSIZE, E068_RBSGMNTSHRNKCNT, E069_RBSEGWAITPCTCNT, E070_PQSERVRSBUSYPCT, E071_PQSRVHIGHWTRPCT, E074_PQQUERYRATE, E075_RCRSVCURSRRATIO, E076_PQRANGESCANPCT, E082_SESSHIGHWATRCNT, E083_DBWRCKPTRATE, E085_TRANSACTIONPCT, E087_PROCESSPCT, E089_ENQUEUEPCT, E090_DSPTCHRBUSYPCT, E091_NUMDSPTCHRCLNTS, E092_SHRSRVRREQWTPCT, E093_SHAREDSERVERPCT, E094_SESUGAMEMCURPCT, E095_SESUGAMEMMAXPCT, E096_SHRDSRVHWMPCT, E097_DISBLDTBLLCKNUM

Sybase Graphing Metrics for use with DBSPI_SYB_GRAPH:

INSTANCENAME, S001_UTILBYCPUPCT, S003_CONTXTSWITCHPCT, S004_TRANSLOGFULLPCT, S007_CONNECTUSERSPCT, S008_TRANSACTNVOLUME, S009_IDXPAGESPLITNG, S010_INDEXDEADLOCKNG, S016_SPACEDUSEDPCT, S018_TOTALDISKIO, S021_CHECKPOINTRATE, S027_ACTIVCURSORSCNT, S029_AVGLKSBYPROCCNT, S031_DATACACHEPCT, S032_PROCDRECACHEPCT, S033_DATACACHMISSPCT, S035_CACHEUTILZTNPCT, S036_LRGEIODENIEDPCT, S037_LRGEIOUTILPCT, S038_APFDENIEDPCT, S039_FULLULCFLUSHPCT, S041_PACKETSSENT, S042_PACKETSRECEIVED, S044_AVGPACKSIZESNT, S045_AVGPACKSIZERCVD, S046_HIGPRIOCHNGSPCT, S047_MEDPRIOCHNGSPCT, S048_LOWPRIOCHNGSPCT, S050_CPUAPPLOGNPCT, S051_IOAPPLOGNPCT, S053_LOCKSUSERPCT, S054_OPNOBJCTSUSDPCT, S055_BLOCKDPROCESSES, S057_DEADLOCKCNT, S061_CONECTPERUSRCNT, S062_LOGSEMAWAITPCT, S063_FREECHKPNTSRATE, S064_WORKRPROCESRATE, S065_PARALLELQRYRATE, S066_ESQREQUESTRATE, S067_WRKRPRCRQDENPCT, S068_WRKRMMRQFAILPCT, S069_PQRUNTADJUSTPCT, S070_HEAPINSERTPCT, S071_NCIXREQMNTIUPCT, S072_NCIXREQMNTDLPCT, S073_SPINLOCKCONTPCT, S074_BUFFGRABDRTYPCT, S075_COMPLTEDSKIOPCT, S076_CACHEHITMISSPCT, S080_BFERWASHINIOPCT, S081_BFERWASHDRTYPCT

User Defined Metrics and Generic Datasources

The UDM numbering range is 700-799 and continues with that range for Reporter and OpenView graphing products. However, for the generic datasource, the Database SPI numbering range for UDMs is 700-798. The “799” number is omitted so that INSTANCENAME column can be included in the UDM generic datasource. A generic UDM datasource, such as DBSPI_ORA_UDM, can contain only metrics 700-798.

Multiple Instances and Generic Datasources

As stated previously, generic datasources also reserve a column for the database instance name, labeled INSTANCENAME. This column, then, contains the information that differentiates the data collected for each instance.

A single generic datasource can contain data from multiple instances. Previous Database SPI versions did not permit multiple instance data to exist in a single datasource; for example, in the past and continuing with OpenView Reporter, if system Kimball had two Oracle instances (Openview and Test), two specific datasources were created, ORADB_KIMBALL_OPENVIEW, ORADB_KIMBALL_TEST.

Contrasting with that method, a generic datasource, like DBSPI_ORA_GRAPH, can store data for multiple Oracle instances; for example, Openview and Test. The data from Openview instance and Test instance would be located in the same generic datasource, only an INSTANCENAME column allows identification of the Oracle instance where the data originated. Likewise, all Oracle reporting metrics would be contained in the DBSPI_ORA_REPORT generic datasource, using the same INSTANCENAME column to uniquely identify the source.

Whenever the Database SPI creates an older version type datasource, it automatically creates the corresponding newer version generic datasource. For example, when ORADB_KIMBALL_OPENVIEW datasource is created (which contains graphing metrics), the DBSPI_ORA_GRAPH generic datasource is also created.

Datasources for User Defined Metrics

The UDM numbering range is 700-799 and continues with that range for Reporter and OpenView graphing products. However, for the generic datasource, the Database SPI numbering range for UDMs is 700-798. The “799” number is omitted so that INSTANCENAME column can be included in the UDM generic datasource. A generic UDM datasource, such as DBSPI_ORA_UDM, can contain only metrics 700-798.

Using DB-SPI with Reporter

If you use HP OpenView Reporter, you can configure the Smart Plug-in for Databases to work with it. From a Windows client system, Reporter can generate reports from DB-SPI data. With the Database SPI, you receive a report package that integrates the two products.

Once integrated with Reporter, the Database SPI adds reports that cover the *availability*, *size*, and *workload* of your database. Automatically generated every night, these Web-based reports provide you with a routine means of checking the health and efficiency of your database. By consolidating information, available otherwise only in pieces, DB-SPI’s integration with Reporter provides you with a more complete view of how your database is performing and growing over time.

In this chapter the following topics are covered to show you how to integrate, and use the DB-SPI Reporter package (you can find other relevant information within the Reporter documentation set, which includes an online *Concepts Guide*, online Help, and Release Notes):

- Enable Reporter Integration on managed nodes
- Complete DB-SPI Reporter Integration on the Windows client (running Reporter)
- Overview of Reporter Metrics from DB-SPI

Install the Report Package

To install from the HP OVO Report Package DVD:

1. At the Windows system running OpenView Reporter, insert the *HP OpenView Smart Plug-ins: Reporting/Graphing Packages for Windows* DVD in the DVD-ROM drive.
2. Go to: `\OV_Reporter\DB_SPI_10.40\`

3 *For OV Reporter 2:*
- Run **DBSPI_Reporter2.exe**
. Files are extracted to `c:\rpmtools\dbspi_install` (default)
- Go to `c:\rpmtools\dbspi_install\Disk1` and run **setup.exe**

For OV Reporter 3
- For Informix reports, run: **DBSPI-INF-Reporter.msi**
- For MS SQL reports, run: **DBSPI-MSS-Reporter.msi**
- For Oracle reports, run: **DBSPI-ORA-Reporter.msi**
- For Sybase reports, run: **DBSPI-SYB-Reporter.msi**

Enabling Reporter Integration on the Managed Nodes

Prerequisite: It is assumed that you have installed the latest DB-SPI version and followed the instructions in Chapter 2, where you deploy software, configure database connections, and assign/distribute templates to the targeted nodes.

This chapter provides information on how to integrate DB-SPI with HP OpenView Reporter. When you complete the instructions here, you will see reports generated by Reporter every night that show information on the monitored database(s).

Enabling Reporter for Use with OVO

In this section you use a DB-SPI application to enable data collection on the managed node. Then (as necessary) you distribute DB-SPI Reporter data collection definitions to the targeted node(s) by assigning and distributing the templates.

Task 1: Enable Reports for the Managed Node (if needed)

If you enabled reports and graphs after configuring the DB-SPI connection to the database on the managed node, you can skip the task below.

1 (Optional) Use the **Customized Startup** option in OVO to add **`-mwrestart`** to the Additional Parameters box of the DBSPI Enable Reports icon if you would use OpenView Performance Agent and would like it to restart automatically when you enable reports on the managed node.

2 Open the Node Bank window and select the node for which you want reports generated in Reporter.

3 Open the Application Bank window and double-click the **DBSPI→Admin** or the **DBSPI→Admin Windows** group to display all administrative applications.

4 Drag and drop the targeted managed node on to the **Enable Reports** application, or with the managed node still selected, double-click the application.

Task 2: Assign/Distribute the Reporter templates

If you assigned the *Quick Start* group to the managed node, you have already automatically deployed Reporter templates and you can skip this task. Otherwise, assign the template(s) to the managed node(s) as instructed.

1 Open the OVO Node Bank window and select the desired managed node(s) designated to run DB-SPI.

2 To assign (select) the templates for distribution from the Actions menu, select **Agents→Assign Templates...**.

3 In the Define Configuration window click **Add...**.

4 In the Add Configuration window click **Open Template Window...**.

5 In the Message Source Templates window, double-click the template group (DBSPI-Oracle, -Informix, -MSS6, -MSS7, or -Sybase) to display the subgroups and Quick Start underneath.

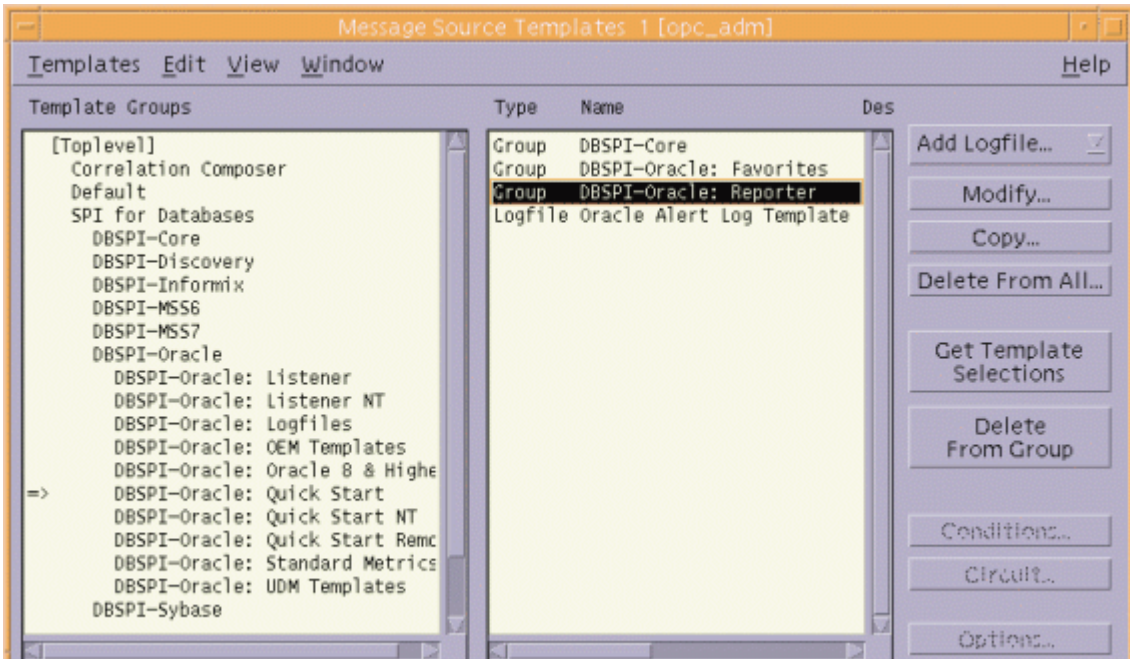
6 Select the template group for the database type you are monitoring:
DBSPI-Oracle: Reporter
DBSPI-Oracle: Reporter NT
(for Oracle on NT)
DBSPI-Informix: Reporter
DBSPI-Sybase: Reporter
DBSPI-MSS

7 In the Add Configuration window select **Get Template Selections**.

8 Click **OK**.

For detailed instructions on OVO template assignment, please refer to the OVO online Help or the *OVO Administrators Task Guide*.

Figure 22 The DB-SPI-Oracle: Reporter templates are selected here



(The templates are assigned [selected] for distribution. Next, you distribute them to the managed nodes.)

9 To distribute the assigned templates, from the Actions menu select **Agents→Install/ Update SW & Config...**.

10 In the Install/Update OVO Software and Configuration window check the **Templates** check box.

11 Select the **Nodes in List** button.

If you did not previously select a node, click the **Get Map Selections** button to list the target node(s) and click **OK** to distribute templates to the managed node(s).

The following message is displayed in the Message Browser:

```
The following configuration information was successfully distributed:
Templates
```

12 *(Skip this step if you added* ***-mwrestart*** *to the Additional Parameters box of the Enable Reports icon in Task 1, step 1, or if you are using the OVO subagent instead of OV Performance Agent [MeasureWare Agent].)*
Restart OpenView Performance Agent servers on the managed node by using the following command:
UNIX: **`mwa restart server`**
Windows:Use the OpenView Performance Agent (MeasureWare Agent) graphical user interface.

The DB-SPI Reporter templates are now distributed to the selected managed node(s). DB-SPI monitors for these templates run every 5 minutes or daily, depending on the template.

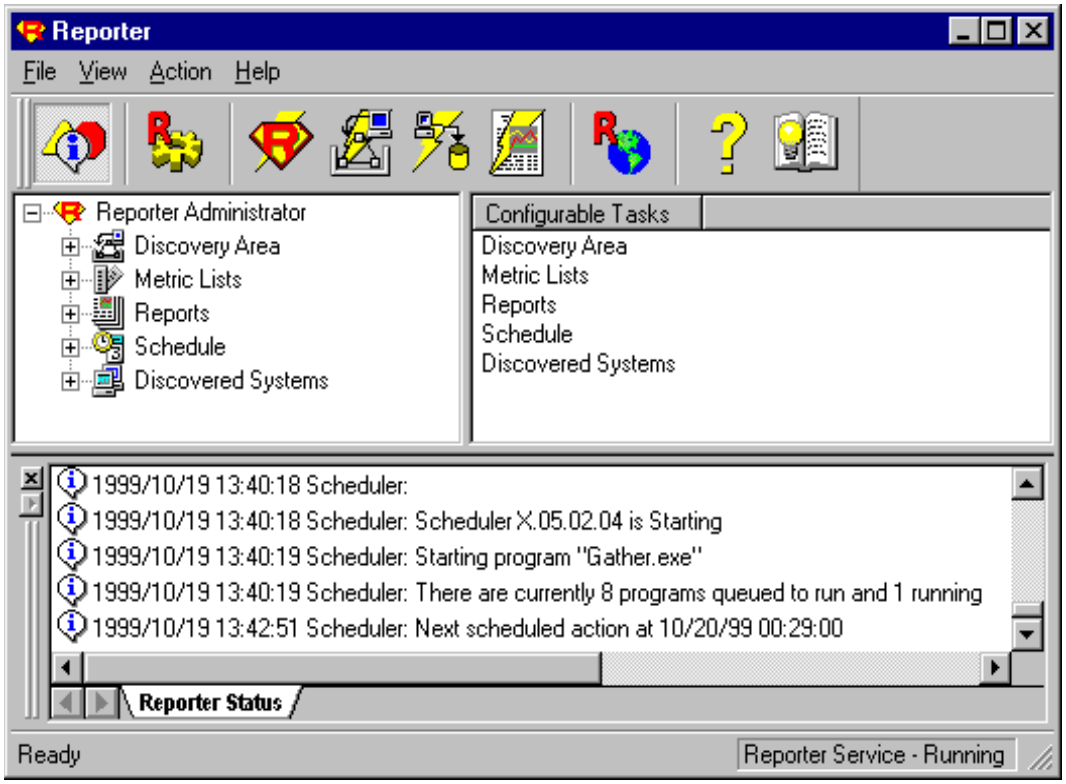
Task 3: Assign Reports

You can find instructions in the Reporter Help for assigning DB-SPI reports to the targeted nodes. To access Help, select **Reports** or **Discovered Systems** in the left panel of the Reporter main window and right-click it. Select **Report Help** or **Discovered Systems Help** from the submenu that appears. See the topic "To assign a report definition to a Discovered Systems Group."

1 If Reporter OVO reports are configured, add DB-SPI OVO Reports by assigning them to the ALL group *Groups and Systems* in the Reporter main window. (See the database specific *Reference Guide* for a list of OVO Reports, for the respective database.)

2 Add group and single system reports by assigning reports as desired. (See the Reporter Help and *Concepts Guide* for complete information.)

Figure 23 The status pane (at the bottom of the Reporter window) shows you information on programs running and errors occurring. You can check the status pane to see that Reporter has been updated with the DB-SPI report templates



Task 4: Resolve System Names for Group and/or Single System Reports

When a monitored database runs on a managed node known by more than one name, you must address the situation in order to generate group and/or single system DB-SPI reports. These reports require that the system name used by the DB-SPI match the name appearing in the Reporter Discovered Systems list.

If you do not need single or group system reports, Reporter can still generate the All systems report, whether or not a system has multiple names. No additional configuration is necessary.

Sometimes system names are shortened (as in an alias), where system **abc.xyz.com** (discovered by Reporter) is also known as system **abc**. System names can also be changed in an environment using Network Address Translation (NAT), where the name is routinely altered when transmitted across a firewall.

To address these situations, you must enter the Reporter (Discovered Systems) name in the DB-SPI `defaults` file. In this file you insert the line:

`SYSTEMID` <*system_name*>

where the *system_name* matches the one contained in the Discovered Systems list. The `defaults` file may or may not already exist. If it does not, you will need to create it.

To update or create the file:

1 At the OV managed node use a text editor to open an existing file labeled `defaults`, or create a new one. For the location of this file, see Data, Log, and Configuration Files on page 238.

2 Add the following single line to identify the system:
`SYSTEMID <system_name>`

3 Save the file.

4 Repeat for each managed node.

Reporter Metrics & Reports

After you have completed the installation and configuration of the DB-SPI Reporter integration (as described in the preceding sections), Reporter can generate reports, based on DB-SPI data.

The first set of reports is generated after Reporter runs through its first nightly schedule. From that point on, you can expect to see updated reports every day because Reporter, by default, re-generates reports every night with the particular day's data.

Refer to the respective DB-SPI *Reference Guides* for details about the DB-SPI reports for Oracle, Microsoft SQL Server, Informix, and Sybase, that are available in Reporter.

The following table lists the active and history message reports and their descriptions.

Table 20 Active Message/History Message Reports/Descriptions

OVO Reports	Description
DBSPI Active Messages	Database SPI-generated, unacknowledged messages.
DBSPI Active Message Severity	Database SPI-generated unacknowledged messages, sorted by severity.
DBSPI History Message	Database SPI-generated, acknowledged messages.
DBSPI History Message Severity	Database SPI-generated, acknowledged messages, sorted by serverity.
Oracle Active Messages	Oracle SPI-generated, unacknowledged messages.
Oracle Active Message Severity	Oracle SPI-generated, unacknowledged messages, sorted by serverity.
Oracle History Messages	Oracle SPI-generated, acknowledged messages.
Oracle History Message Severity	Oracle SPI-generated, acknowledged messages, sorted by serverity, that have been acknowledged.
Informix Active Messages	Informix SPI-generated, unacknowledged messages.
Informix Active Message Severity	Informix SPI-generated, unacknowledged messages, sorted by serverity.
Informix History Messages	Informix SPI-generated, acknowledged messages.
Informix History Message Severity	Informix SPI-generated, acknowledged messages, sorted by serverity, that have been acknowledged.
Sybase Active Messages	Sybase SPI-generated, unacknowledged messages.
Sybase Active Message Severity	Sybase SPI-generated, unacknowledged messages, sorted by serverity.

Table 20 Active Message/History Message Reports/Descriptions

OVO Reports	Description
Sybase History Messages	Sybase SPI-generated, acknowledged messages.
Sybase History Message Severity	Sybase SPI-generated, acknowledged messages, sorted by serverity, that have been acknowledged.
SQL Server Active Messages	SQL Server SPI-generated, unacknowledged messages.
SQL Server Active Message Severity	SQL Server SPI-generated, unacknowledged messages, sorted by serverity.
SQL Server History Messages	SQL Server SPI-generated, acknowledged messages.
SQL Server History Message Severity	SQL Server SPI-generated, acknowledged messages, sorted by serverity, that have been acknowledged.

Modifying Reporter Metric Collections

The DB-SPI collected data is sent to Reporter for report generation at 11:45 PM every night. To change any aspect of the Database metric collections, do so in the usual way by opening a template and making changes within it. You can change Reporter *space management* and *workload metric* parameters in the collector template's command line, entered in the **Command** text box.

You can find the Reporter template group as follows:

1. Open the database application group; for example, you would open **SPI for Databases→DBSPI-Oracle**
2. Select **DBSPI-Oracle: Reporter**.

You can change the metric parameter (set in the collector template) after the colon (**:**), following the metric ID; for example `dbspicao -c XXX -m 210:1`. The "1" that follows `-m 210:` indicates that the metric value must equal at least 1MB before it is collected and reported on.

DB-SPI Reporter templates are as follows:

Table 21 Database SPI collector templates for reports

Oracle Templates	Command Line	Notes
DBSPI-Ora-1d-Reporter	dbspicao -c XXX -m 210:1,212:1,215:1,213:0	See (2)
DBSPI-Ora-05min-Reporter-NT	dbspicao -c XXX -m 201	See (1)
DBSPI-Ora-1d-Reporter-NT	dbspicao -c XXX -m 210:1,212:1,215:1,213:0	See (2)
Sybase Templates	Command Line	Notes
DBSPI-Syb-05min-Reporter	dbspicas -c XXX -m 225	See (1)

Table 21 Database SPI collector templates for reports

DBSPI-Syb-1d-Reporter	dbspicas -c XXX -m 222:1,223:1,224:1,226:0	See (2)
Informix Templates	Command Line	Notes
DBSPI-Inf-05min-Reporter	dbspicai -c XXX -m 201	See (1)
DBSPI-Inf-1d-Reporter	dbspicai -c XXX -m 216:1,218:1,221:1,245:0	See (2)
MS SQL Server template	Command Line	Notes
DBSPI-MSS-05min-Reporter	dbspicam -c XXX -m 243	See (1)
DBSPI-MSS-1d-Reporter	dbspicam -c XXX -m 240:1,241:1,242:1,244:0	See (2)

Table Notes:
(1) Runs availability metric. No changes should be made to this template.

(2) Runs space and I/O metrics. For space metrics the metric parameter is the smallest size object in MB allocated that will be included in the collection/report. For I/O metrics the metric parameter is the smallest number of I/Os for an object that will be included in the collection/report.

For more details on making changes to templates, please see the *OVO Concepts Guide* for instructions.

Using DB-SPI with OpenView Performance Insight

HP OpenView Performance Insight includes reports that are customizable and that can be provisioned to break out data by an individual datasource element (database instance) or grouped elements (database metrics of the same type). Further, log-in security can limit the information, such as specific customer data or department data, that a user can see. The types of reports generated by HP OpenView Performance Insight are:

- The Executive Summary reports, which give an overall view of a group of elements. These reports show volume and other metrics of interest.
- The Capacity Planning report provides a view of under and over-utilized elements. The information in these reports help in determining how to balance loads and avoid costs by removing the need to purchase new resources.
- The Forecast report focuses on resources that are projected to become over-utilized in 90 days. This report gives administrators time to plan upgrades before bottlenecks affect users.
- The Hot Spot, Quick View and Top Ten reports identify elements of possible concern by exceptions, degree of change and other criteria.
- The Service Level report shows the elements with the worst availability and worst response time—the two leading metrics used in SLAs.

Additionally, HP OpenView Performance Insight includes forms for creating read/write reports, allowing you to change descriptions or provision new objects. All reports are customizable and can be viewed through a web browser, e-mailed, or downloaded into PDF, CSV or HTML files.

How OpenView Performance Insight Uses DB-SPI Data

OpenView Performance Insight uses data stored in the Database SPI generic datasources (please see Table 19 on page 82 for examples of generic datasources). The Database SPI generic datasources are named according to the four database types: Informix, MS SQL Server, Oracle, or Sybase and also according to the metric type, which in this case is reporting and are as follows:

- DBSPI_ORA_REPORT
  (OVPI templates currently exist only for Oracle)
- DBSPI_INF_REPORT
- DBSPI_MSS_REPORT
- DBSPI_SYB_REPORT

(Other datasources store metrics for graphing, and user-defined [metrics] (UDMs).

Please refer to the OpenView Performance Insight documentation for integration information.

Using DB-SPI with HP OpenView Graphing Products

Graphing capabilities are available with the separately purchased products:

- **HP OpenView Performance Manager (OVPM)**
  (works with both new and existing OpenView agents)
- **HP OpenView PerfView** (works only with OpenView Performance Agent (also known as MeasureWare Agent)

From the OVO Message Browser, you can generate graphs from most DB-SPI alarm messages by pressing the *Perform Action* button from the message details. This *operator action* displays a graph of the metric causing the alarm and other related metrics.

Configuring OpenView Performance Manager

OpenView Performance Manager (OVPM) is compatible with both new and existing OpenView performance subagents/agents. So, whether you use the new OpenView performance sub-agent or the previous OpenView Performance Agent (also known as MeasureWare Agent), you can generate graphs.

If you configure OpenView Performance Manager (as described below), execution of an operator action (in the OVO console) will launch your configured browser, displaying OpenView Performance Manager. If you do not configure OpenView Performance Manager, executing an operator action will launch PerfView.

Task 1: Install the OVPM Graphing Package

For OpenView Performance Manager on Windows: Refer to the table below for installing the OVPM integration software on the Windows.

Table 22 OVPM Integration Packages

Purpose	DVD-ROM	/Directory/ /File
OVPM integration for Informix	OV Reporter Packages DVD-ROM	/OV_PM/DB_SPI_10.40/ DBSPI-INF-OVPM.msi DBSPI-INF-OVPM_jp.msi (Japanese)
OVPM integration for Sybase	OV Reporter Packages DVD-ROM	/OV_PM/DB_SPI_10.40/ DBSPI-SYB-OVPM.msi DBSPI-SYB-OVPM_jp.msi (Japanese)
OVPM integration for Oracle	OV Reporter Packages DVD-ROM	/OV_PM/DB_SPI_10.40/ DBSPI-ORA-OVPM.msi DBSPI-ORA-OVPM_jp.msi (Japanese)
OVPM integration for MS SQL Server	OV Reporter Packages DVD-ROM	/OV_PM/DBSPI_10.40 DBSPI-MSS-OVPM.msi DBSPI-MSS-OVPM._jp.msi (Japanese)

For OpenView Performance Manager on UNIX: Copy files to the separate OVPM system (but skip task if OVPM is on an OVO management server system).

When you installed the Database SPI on the OVO management server, you installed files for OpenView Performance Manager (OVPM) as well. Successful graph generation requires that you identify which of the three scenarios fits your installation, then proceed accordingly: The first scenario requires no action, while the other two require that you copy files from one system to another.

The number of files on the OVO management server depends on whether you installed one or more Database SPI types; if you installed only Oracle, you will see the Oracle (`ORA`) graph files; if you installed other Database SPI types, such as Informix (`INF`), MS SQL Server (`MSS`), or Sybase (`SYB`), you will see those graph files. See the list in Scenario #2 and #3 for all possible files.

Scenario #1 — English/Japanese environments, where the OVO management server and OVPM are installed on the same system: No further installation is necessary.

Scenario #2 — English environments, where the OVO management server and OVPM are installed on separate systems: Copy the OVPM files from the OVO management server directories to the identical directories of the OVPM UNIX system. Directories/files are:

```
/opt/OV/newconfig/OVPM:
  DBSPI_FormTemplate.txt
  DBSPI_FormTemplate_jp.txt
  DBSPI_FrameTemplate.txt
  DBSPI_FrameTemplate_jp.txt
```

```
  DBSPI_GraphsINF.txt
  DBSPI_GraphsMSS.txt
  DBSPI_GraphsORA.txt
  DBSPI_GraphsSYB.txt
  DBSPI_OvowGraphsINF.txt
  DBSPI_OvowGraphsMSS.txt
  DBSPI_OvowGraphsORA.txt
  DBSPI_OvowGraphsSYB.txt

/opt/OV/OVPM/webapps/OVPM:
  empty.htm
```

Scenario #3 — Japanese environments, where the OVO management server and OVPM are installed on separate systems: Complete steps as follows.

1 Copy the OVPM files from the OVO management server directories to the identical directories on the OVPM UNIX system. Directories/files are:

```
/opt/OV/newconfig/OVPM:
  DBSPI_FormTemplate.txt
  DBSPI_FormTemplate_jp.txt
  DBSPI_FrameTemplate.txt
  DBSPI_FrameTemplate_jp.txt
  DBSPI_GraphsINF.txt
  DBSPI_GraphsMSS.txt
  DBSPI_GraphsORA.txt
  DBSPI_GraphsSYB.txt
  DBSPI_OvowGraphsINF.txt
  DBSPI_OvowGraphsMSS.txt
  DBSPI_OvowGraphsORA.txt
  DBSPI_OvowGraphsSYB.txt

/opt/OV/OVPM/webapps/OVPM:
  empty.htm
```

2 Move the two files whose names end in "_jp" from the `/opt/OV/newconfig/OVPM` directory to its subdirectory: `/opt/OV/newconfig/OVPM/jpn`

3 With the files now relocated to the `/jpn` directory, change the file names to remove the "_jp" so that the directory now shows files as follows:

```
/opt/OV/newconfig/OVPM/jpn:
  DBSPI_FormTemplate.txt
  DBSPI_FrameTemplate.txt
```

Task 2: (as necessary) Enable Graphs.

To use either OpenView graphing product, you must enable graphs. You may already have done so when you configured the DB-SPI connection to databases on the managed nodes. On saving that file, if you chose not to accept the default to enable graphs, you can do so now.

To enable graphs:

1. Open the Application Bank window.
2. Select:
   DB-SPI→Admin or
   DBSPI→Admin Windows
3. Run **Enable Graphs** for each managed node.

Task 3: Configure the Web browser.

The Database-SPI uses the **ovweb** utility to start your web browser for displaying graphed metrics. Before **ovweb** can launch your Web browser, however, you must insert the browser invocation command within the `ovweb.conf` file. If no browser invocation command is included in the `ovweb.conf` file, **ovweb** tries to start Netscape (which must be version 4.7 or higher).

The `ovweb.conf` file must be located in the directory specified by the environment variable `$OV_CONF`, which is used by OpenView Operations. If you need to view this information, open the `/opt/OV/bin/OV/envvars.sh file` and look for the `$OV_CONF` definition.

- **Enter the browser invocation command in the `ovweb.conf` file:** The browser invocation command must contain a `%s` to allow the Database SPI to pass a URL to the browser. Open the file and insert the command according to the entry syntax and example as follows:

  Syntax: `Browser:` <*browser command*> %s
  Example: `Browser: /opt/netscape/netscape %s`

- **Check the Web browser setting:** To display the DB-SPI graphs, ensure that your browser is *Java Script enabled*. Check the setting within the browser's Preferences.

  For information about launching a browser in OpenView Operations, consult the man pages for **ovweb** and `ovweb.conf` and `ov.envvars`. To access instructions, at a command prompt enter: `man ovweb`

Task 4: Configure the OpenView Performance Manager system.

1. On the OVO management server, open the Application Bank window and select the **DBSPI→Admin** group.
2. Double-click **Configure Graphs** to open the configuration file (with your default text editor): `/opt/OV/DBSPI/bin/ovpm.cfg`
3. Enter the (UNIX or Windows) system URL where OVPM is installed:
   Syntax: `GRAPH_URL http://`<*server*>`:<port>/OVPM/index.html`
   Example #1: `GRAPH_URL http://server1:8080/OVPM/index.html`
   Example #2: `GRAPH_URL http://server2/HPOV-Reports/ovpm.htm`

   If you previously configured graphs for OVPM 4.x (Windows system, as shown in the Example #2 above), you do not need to revise your entries. The following is still accepted: `OVPM_SERVER` <*system_IP_address*> or <*system_name*>
   Example: `OVPM_SERVER mywinsystem.rose.hp.com`
4. Save the file.

Task 5: Configure DB-SPI managed nodes (as necessary)

If OpenView Performance Manager (OVPM) and OpenView Reporter are installed on the same system, and DB-SPI managed nodes are among Reporter's discovered systems, you can skip this task.

If, however, OVPM is installed on a standalone system, you must complete the following.

1 On the system where OpenView Performance Manager is installed, open the text file:
*<*install_directory*>\Data\systemsCODA.txt
or
**<*install_directory*>\Data\systemsMWA.txt

*The systemsCODA.txt file includes all systems where DB-SPI data is stored using the OpenView Operations performance subagent.

**The systemsMWA.txt file includes all systems where DB-SPI data is stored using the HP OpenView Performance Agent, also known as MeasureWare Agent.

2 Insert all DB-SPI managed node names in the file:
<*system_name_1*>
<*system_name_2*>
<*system_name_3*>

Example (two system names):
```
lani.cal.hp.com
annie.col.hp.com
```

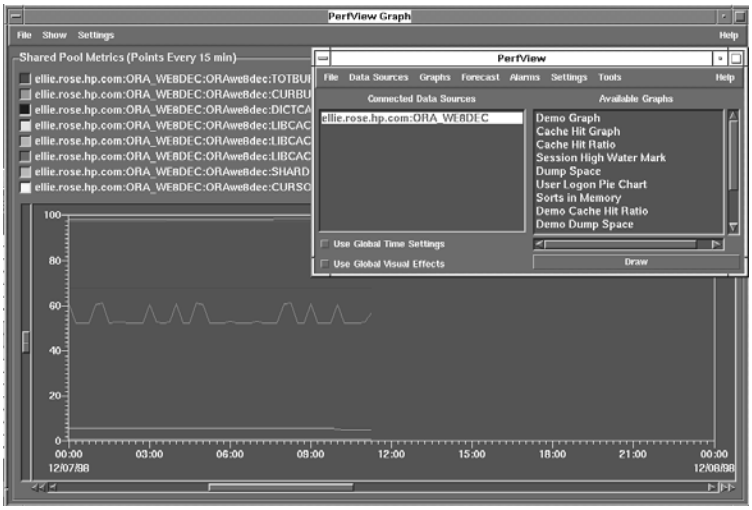
Configuring OpenView PerfView

PerfView works with OpenView Performance Agent (also known as MeasureWare Agent). If you use the OpenView Performance Agent, the DB-SPI collector/analyzer writes data to the `.dat` file(s) as it collects the metric data in the following manner:

- ***5-minute collection interval.*** Data points are updated every 5 minutes (the data will be 5 minutes behind the data collected).
- ***15-minute collection interval.*** 3 consecutive data points in MeasureWare (collected every 5 minutes) will be the same.
- ***1-hour collection interval.*** 12 consecutive data points (collected every 5 minutes) will be the same for these metrics.
- ***Daily collection interval.*** Data points change only once every 288 collections.

It is advisable to take the collection interval into account when displaying the DB-SPI metrics in PerfView.

Figure 24 A PerfView graph generated from an Operator Action



Enable Graphs

To use PerfView, you must have enabled graphs. You may have already done so during the Database SPI installation when you configured the connection to the database. When you were prompted to enable graphing. If you chose not to accept the default to enable graphs, you can do so now.

To enable graphs:

1. Open the Application Bank window.
2. Select **DBSPI→Admin** or **DBSPI→Admin Windows→ Enable Graphs**.
3. Run **Enable Graphs** for each managed node.
4. Before the data can be viewed with PerfView, you must *either*:

   a. Restart the OpenView Performance Agent servers on the managed node by using the following command or action:

   `(UNIX)` **`mwa restart server`**
   `(NT)` Use the MeasureWare Agent GUI

   or

   b. Use the Customized Startup option in OVO to add the following to the Additional Parameters box of the **Enable Graphs** icon before execution:

   `-mwrestart`

5. After OpenView Performance Agent re-starts on the managed node and data is logged, you can view graphs with PerfView by selecting the appropriate data source name (according to platform) (please see online Help for more detailed instructions on generating graphs):

   `ORADB_<`*system_name*`>_<`*database_name*`>`
   `INFSV_<`*system_name*`>_<`*server_name*`>`
   `SYBSV_<`*system_name*`>_<`*server_name*`>`
   `MSSQL_<`*system_name*`>_><`server_name`>`

Checking PerfView integration errors

(UNIX) An OpenView Performance Agent *application group* is a set of related processes that log their effect on computing resources like disk, memory, and CPU. For more information, consult the *HP OpenView Performance Agent User's Manual*.

You can view integration errors in the file:

UNIX: `/var/opt/OV/dbspi/log/mw_int_status`
Windows: If errors occur, they are displayed in the window.

Metrics for Graphing and/or Alarming

Some metric data may be used strictly for graphing purposes. You can determine if the metric is a graphing-only metric by viewing the DB-SPI metric summary in the DB-SPI *Oracle Reference, Informix Reference, Sybase Reference,* and *MS SQL Reference* guides, where you can find a column labeled "Graph." The key to entries in the column is as follows:

- G means data is used for graphing only
- A means data is used for alarming only
- G & A means data is used for both graphing and alarming

If this column contains an "A," the metric generates a message whenever the metric value exceeds a threshold (alarm condition).

Using Metrics for Graphing Only/Avoiding Nuisance Alarms

You can avoid alarms/messages if you need the data only for graphing. To prevent alarms or messages from being generated, set the monitor template threshold for the metric to an extreme value (0 for Minimum metrics and 100 or higher for Maximum metrics). For example, for Oracle metric #0022, which measures buffer/cache hit ratio, you might choose to set the threshold to 0.0%. This minimum threshold setting (below which incoming data values would never drop) would allow you to receive and graph data while avoiding nuisance alarms.

Graph Type

The type of information you can receive in a graph is shown below along with the Oracle metrics that can generate these graphs.

Table 23 Graph Types for Oracle Metrics

Description	Metric Number
Redo and Archive	32, 33, 34, 56, 57, 58
Space Management	6, 7, 8, 9, 11, 16, 17, 18
Shared Pool	22, 23, 26, 27, 39, 40, 45, 59
Initialization Limits	28, 31, 85, 87, 89
Errors	77, 78, 79, 80, 81
Rollbacks	54, 67, 68, 69
Space on Dump Devices	62, 63, 64, 65, 66

Table 23 Graph Types for Oracle Metrics

Description	Metric Number
Tables/Indexes	30, 41, 42, 46, 47, 48
Waits/resources	21, 24, 38, 43
Sorts	18, 19, 51, 52
Calls	44, 49, 50, 75
Checkpoints	35, 83
Logon and Sessions	37, 82

Table 24 Graph Types for Informix Metrics

Description	Metric Number
Logfile ChkPtLog	15, 64, 65, 66, 69, 70, 71
Locks	73, 74, 75, 76
Sessions	5
ShMem	32, 33, 36, 37, 39, 40
ShmTot	31
SortScan	47, 51, 52, 53
Txns	56, 57, 58, 59, 60
Writes	42, 43, 44

Table 25 Graph Types for Sybase Metrics

Description	Metric Number
CPU	1, 3, 203
Cache	31, 32, 33, 35, 36
Database	4, 7, 8, 9, 10
Disk	213, 216, 18, 21
Index	70, 71, 72
Memory	37, 38, 39, 73, 74, 75, 76, 80, 81
Network	41, 42, 44, 45
Parallel	64, 65, 66, 67, 68, 69
Process	46, 47, 48, 50, 51, 251
Server	27, 29

Table 26 Graph Types for MS SQL Metrics

Description	Metric Number
Access	51, 52, 53, 54, 55, 56
Cache	19, 20, 22
Error	23, 24, 28
I/O Stats	3, 4, 5, 6, 7, 8, 21
Latches	68, 69, 76
Locks	13, 70, 71, 72, 73, 75
LRU Stats	1, 2
RA Stats	10, 12
Server	14, 17, 25, 29, 64, 74
Trans	9, 66
Users	11, 26, 31

Customizing Operator-Initiated Graphs (PerfView only)

Date ranges and summarization levels can be customized by modifying conditions in the desired message source templates. To do so, add the preferred date range or summarization level in quotes at the end of the Command field.

Specifying Date Ranges

Use the following syntax to specify date ranges:

```
"-range <date_range>"
```

Figure 25 Specifying a Date Range in the Command text box

The date range can be specified as an interval of one day, one week, one month, or one year. All ranges begin at 2:00 AM on the first day of the range interval. One-month intervals begin on the first day of the month, and one-year intervals begin on the first day of the year. The default date range specification is “-range d-3” (which means the past three days).

Two formats can be used for the specifier:

Relative. Use a dash (-) in the range. This format lets you specify intervals relative to the current day. One-week intervals begin on Sunday. For example, "-range w-1" specifies Sunday through Saturday of the last week and "-range d-30" specifies the last thirty days of data. Abbreviations are as follows:
day = d, week = w, month = m, year = y

Absolute. The first day of a one-week interval is on the same day of the week as January 1 for the current year. For example, "-range w1" specifies the first 7 days of the current year, beginning on the day January 1 occurs and "`-range w2`" specifies January 8 through January 14 of the current year. To display data for the last fifteen days, use the following syntax in the Operator initiated Command field:

`/opt/OV/DBSPI/bin/dbspiorapv*` <$MSG_NODE_NAME> <$OPTION(dbname)> sharedpool "-range d-15"

*__For Oracle__, use `/opt/OV/DBSPI/bin/dbspiorapv`
*__For Informix__, use `/opt/OV/DBSPI/bin/dbspiinfpv`
*__For Sybase__, use `/opt/OV/DBSPI/bin/dbspisybpv`
*__For MSSQL__, use `/opt/OV/DBSPI/bin/dbspimsspv`

Specifying Summarization Levels

PerfView can provide data points only for complete summarizations. For example, if one hour of data is not yet available, you cannot generate a graph based on one-hour summarization data points.

Use the following syntax to specify summarization levels:

```
-sum <summarization_level>
```

Valid values for <*summarization_level*> are:

Table 27 Valid Values for Summarization Levels

Abbreviation	Summarization Level
5M	Five minutes
15M	Fifteen minutes (default)
30M	Thirty minutes
1H	One hour
3H	Three hours
6H	Six hours
12H	Twelve hours

For example, to see data points every thirty minutes, enter the following in the Operator initiated Command field to:

```
/opt/OV/DBSPI/dbspiorapv* <$MSG_NODE_NAME> <$OPTION(dbname)>
sharedpool  "-sum 30M"
```

*__For Oracle__, use `/opt/OV/DBSPI/bin/dbspiorapv`
*__For Informix__, use `/opt/OV/DBSPI/bin/dbspiinfpv`
*__For Sybase__, use `/opt/OV/DBSPI/bin/dbspisybpv`
*__For MSSQL__, use `/opt/OV/DBSPI/bin/dbspimsspv`

To change both the date range and summarization, specify the new date range followed by the new summarization level at the end of the Command field. For example:

```
<$OPTION(dbname)> sharedpool  "-range d-15"  "-sum 30M"
```

If you want to change the summarization level or range for all graphs, you can modify the file `dbspixxxpv` directly.

5 Oracle Advanced Customizations

This chapter covers issues specific to Oracle, according to the following topics:

Customizing Oracle Roll-up Metrics

Roll-up metrics for Oracle have two threshold settings: one within the collector template and the other within the monitor template. Explanations of these two settings follow.

Oracle Roll-up and Drill-down Metrics

DB-SPI collects some Oracle metrics that target data for multiple objects. These metrics are called “roll-up” metrics and include: Oracle metrics 3, 6, 8, 9, 11, 16, 17, 18, 31, 38, 69. If a corresponding “drill-down” metric exists that shows a specific occurrence of a roll-up metric, it is the same number within a 200 range; for example, roll-up metric #006 has the corresponding drill-down metric #206.

For Oracle drill-down metrics, you can modify DBSPI-0203, DBSPI-0206, DBSPI-0216, DBSPI-0217, and/or DBSPI-0218.

Roll-up metrics alarm only once for multiple objects. For example, when a message (alarm) is sent to OVO for metric #69, the Message Browser shows that the percentage of the total rollback segment waits to gets is high (by default, are >=5%). Also included in the message text is the database name with the highest percentage value of waits to gets. For example:

```
DBSPI-0069.1: 9% rollback segment with wait percentage too high for
DB_0001 (>=5%>).
```

Oracle metrics 0016, 0017, and 0018 are roll-up metrics that alarm once (with only one message in the Message Browser) for one or more threshold violators. For metrics 0016-0018 the object is the segment name.

Five roll-up Oracle metrics have parallel drill down equivalents. They are 0003, 0006, 0016, 0017 and 0018 with drill-down equivalents 0203, 0206, 0216, 0217 and 0218. For these metrics, instead of sending a single message for multiple objects that have exceeded the threshold set in the collector template, DB-SPI sends one message per object threshold violation to OVO. For these metrics, OVO generates a separate message for each object (tablespace or segment) that exceeds the threshold. In order to threshold on individual databases and objects, DB-SPI sends in the object field for the database and object in the form:

Table 28 Oracle drill-down metrics with database & object

Metric	Object
0206	Database:TableSpace

An example is the best way to illustrate this point. Let's say you want to set up different thresholds for individual segments for Oracle Metric 217:

Table 29 Oracle drill-down metric #217* example

Condition Name&Type	Condition Object Pattern	Threshold	Explanation
DBSPI-0217.1 +Message on Matched Condition	ora733:SEGMENT1	95	Threshold condition for SEGMENT1 of database ora733. Object pattern is case sensitive
DBSPI-0217.2 +Message on Matched Condition	ora733:SEGMENT2	85	Threshold condition for SEGMENT2 of database ora733. Object pattern is case sensitive
DBSPI-0217.3 - Suppress Matched Condition	ora733:SEGMENT1 || ora733:SEGMENT2		If the threshold was not exceeded for these two segments, then you must suppress the message if either of these segments gets past 0217.1 or 0217.2
DBSPI-0217.4 +Message on Matched Condition		80	A blank object pattern to catch all remaining segments

To improve performance, you can include a second threshold with Oracle Metric E217. This threshold acts as a filter which selects segments according to percentage full for retrieving data.

To filter message/alarm generation for Oracle metric E217:

1. Find the collector template that contains the metric (the collector template is always last in the any template group),
2. Highlight the template and select Modify.
3. In the dialog that appears within the `Command` text box find the number 217 (for metric 217) and add a colon and the percentage full as the threshold for generating messages; for example, in the command line below, metric 217 generates messages when it exceeds 80 percent full:
   ```
   dbspicao - c DBSPI-Ora-15min -DrillDown -m 217:80
   ```

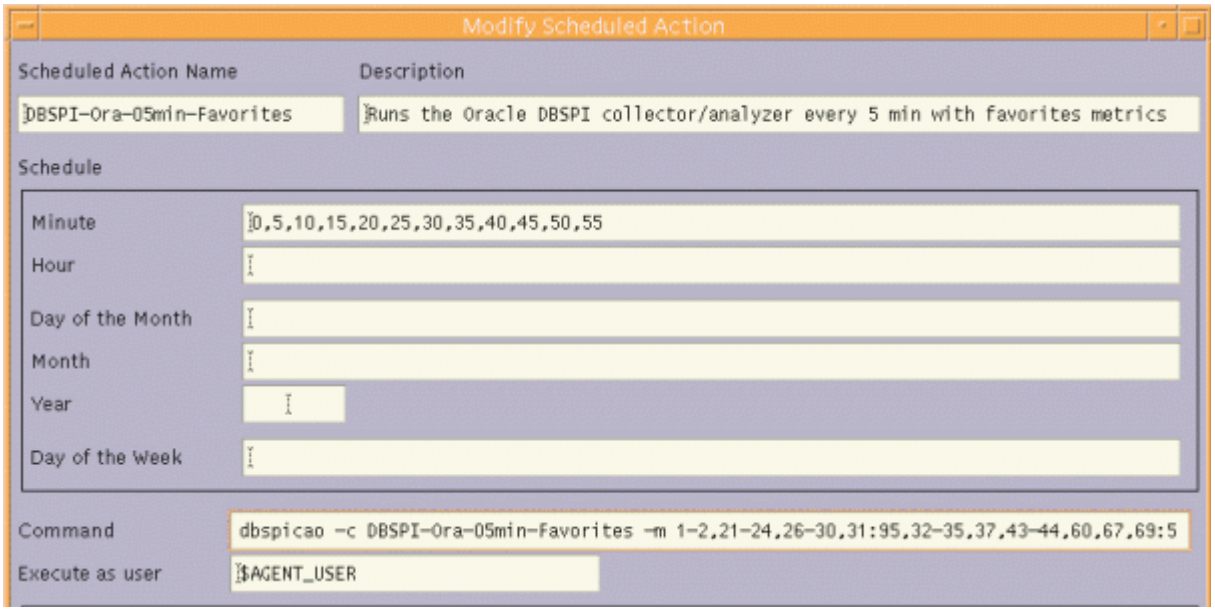
Setting the Collector Template Threshold for an Oracle Roll-up Metric

The collector template threshold sets a minimum or maximum value that when exceeded by individual objects designates a violation (for Oracle metric #6, this threshold is <=10%, meaning that any tablespace with free space less than or equal to 10% is a violation of the threshold).

To set this first threshold for a roll-up metric, you use the command line within the collector template. This threshold is different from the threshold set in the monitor template (which is 0.5, meaning any number greater than zero triggers a violation). The monitor template threshold counts the number of violations, while the collector template sets a threshold for each object. For example, the command line in the collector template `DBSPI-Ora-05min-Favorites` would look like this (see next page):

```
dbspicao -c DBSPI-Ora-05min-Favorites -m 1-2,
21-24,26-30,31:95,32-35,37,43-44,60,67,69:5,83
```

Figure 26 Command Line with Oracle Roll-up Metric 69



For the Oracle roll-up metrics (8, 9, 11, 17, 18, 31, 38, 69), the true thresholds are shown on the command line (for example, metric #69 shows the threshold as <=5).

Oracle metrics **3**, **6**, **206**, and **216** (*DBSPI-0003, DBSPI-0006, DBSPI-0206,* and *DBSPI-0216*), are roll-up metrics not included in any DB-SPI metric collections. To collect data for any of these metrics, insert its number in a collector template `Command` text box. Each metric has a command line default threshold of "1." To change the default, include a colon followed by a value after the metric number, for example, `3:4`.

Setting the Monitor Template Threshold for an Oracle Roll-up Metric

The Oracle monitor template for metric #69 (DBSPI-0069) has a threshold that addresses the COUNT of violations of the collector template threshold. The monitor threshold value is set to 0.5, meaning that any value greater than or equal to (>=) 0.5 causes a message to be sent to the Message Browser.

Oracle roll-up metrics have their true thresholds on the command line (in the collector template); the number of objects exceeding that threshold are counted and the count threshold is set in the monitor template. When the count threshold is exceeded, a message is sent to OVO.

DB-SPI Collector Template Command Line Customization For Oracle Roll-up Metrics

Roll-up metrics thresholds are defined in the command line area in collector templates. Collector templates group DB-SPI metrics according to collection interval (5-minute, 15-minute, one-hour, or one-day). In the Oracle-Favorites group, collector templates are always the last templates within the group. Within these collector templates you can use the command line to alter "roll-up" metric thresholds. A "roll up" metric should have a colon (:) following its metric number in the collector template's command line area. For example:

```
dbspicao -c DBSPI-Ora-05min-Favorites -m 1-2,
7,9:20,21-24,26-35,37,43-44,60,67,69:5,83
```

You can see that metrics #9 (9:20), and #69 (69:5) are all roll-up metrics.

You can use the **dbspicao m** command to change the threshold. For Oracle, metrics 8, 9, 11, 17, 18, 31, 38, 69 are all roll-up metrics that require a command line threshold to be set.

To modify the command line threshold, follow these steps:

1. Select the **Message Source Templates** window.
2. Select the desired metric group, then double-click to display all monitor templates.
3. Select the collector template that executes the desired metric and click the **Modify** button. (For example, for Oracle Quick Start, you would select SPI for Databases→DBSPI-Oracle→DBSPI-Oracle Quick Start→Favorites→Ora-5-min, 15-min, 1-d(ay), or 1-h(our).)
4. In the **Command** text box, find the metric to modify. (Metrics are in the form <*metric_number*>:<*threshold_value*> `as in` **16:5**. In this case, metric 16, which calculates the number of extents available in a tablespace, has its threshold value set to <=5.)

The templates containing metrics using command line thresholds within the collector templates are shown below:

Table 30 Oracle templates with command line thresholds

Oracle Template Name	Metrics with Customizable Collector Template— Command Line Thresholds
DBSPI-Ora-05min DBSPI-Ora-05min-SQLNet	9,11,31,38,69
DBSPI-Ora-15min DBSPI-Ora-15min-Favorites DBSPI-Ora-15min-SQLNet	16,17, 18
DBSPI-Ora-1h DBSPI-Ora-1h-Favorites	8, 57*

Table 30 Oracle templates with command line thresholds

Oracle Template Name	Metrics with Customizable Collector Template— Command Line Thresholds
DBSPI-Ora-1d DBSPI-Ora-1d-Favorites DBSPI-Ora-1d-SQLNet	42**
DBSPI-Ora-05min Favorite	9,31,69
DBSPI-Ora-1h-SQLNet	8

* = For 57, average time between archive writes for past N days.
** = For 42, the number of days since object last analyzed.

An example of a modified command line is provided below:

```
dbspicao -c DBSPI-Ora-05min -m 1-2,
7,9:15,11:100,20-24,26-35,37,39-41,43-45,48-51,54,59-60,67,69:5,75,83
,85,87,89
```

Customizing Tablespace Metrics

The Database SPI for Oracle includes three pairs of roll-up/drill-down metrics that help you monitor tablespaces by looking at available segments and extents. These metrics rely on complex calculations that take into account the variety of ways in which Oracle manages database space. According to Oracle tablespace management features, the metrics return information as follows:

- *Metric 003*: number of tablespaces that cannot extend — because of insufficient extents
  ***Quick Start Metric 203*:** specific tablespaces that cannot extend
- *Metric 006*: number of tablespaces with a low percentage of free space
  Metric 206: specific tablespaces with low percentage of free space
- ***Quick Start Metric 016*:** number of segments that cannot extend — because of insufficient extents
  Metric 216: specific segments that cannot extend

Some auto-extend datafile types Support

Oracle metrics 3, 203, 6, 206, 16, and 216 need to be able to determine the capacity of file systems or disks when datafiles have auto-extend enabled. These metrics do not perform correctly with some types of auto-extend datafiles. See the table below for supported file systems (FS) and raw disks with or without logical volume manager (LVM).

yes: supported
no: not supported
--: not applicable.

Table 31 Supported/non-supported Data File Types

Auto-extend Data File Type					
Operating System	Default FS (not RAW)	VERITAS FS (not RAW)	Oracle CFS (not RAW)	RAW disk using LVM	RAW disk not using LVM
HP-UX	yes	yes	yes	yes	no*
Solaris	yes	yes	--	yes	no*
Linux	yes	yes	yes	no*	no*
AIX	yes	yes	--	yes	no*
Windows	yes	--	yes	--	yes
Tru64	yes	--	--	no*	no*

*Each 'no' designation means that free disk space is monitored, but with no regard to auto-extend. Specifically, whenever the Database SPI encounters an object sitting on a RAW disk with auto-extend on, a warning is generated indicating that even though auto-extend is set to ON, the Database SPI monitors free allocated disk space as though auto-extend were OFF.

Quick Start & Other Database Space Metrics

Metric 203 and Metric 016 are the only database space metrics included in the DBSPI-Quick Start group because they effectively cover tablespace issues from differing perspectives. Both alert you to situations involving insufficient extents: one with information on *specific tablespaces* unable to extend, and the other with information on *segments* unable to extend.

The sections that follow cover these and the remaining Database SPI tablespace metrics so that, if needed, you can knowledgeably modify those in Quick Start and/or add those not included. The metric data calculations are shown for information purposes only.

Metric 003 and Quick Start Metric 203: Free Tablespace Available

Metrics 003 and 203 both focus on tablespaces running out of free space. However, metric 003 looks at tablespaces with free extents low, while metric 203 looks at the number of extents needed for the largest next extent of a segment (max(NEXT_EXTENT) from DBA_SEGMENTS). Because of these differences, Metric 003 has a maximum threshold—alarming with any increase in tablespaces with free extents low (one or more), while Metric 203 has a minimum threshold—alarming when the number of extents drops below the acceptable level (threshold setting).

Metric 003 and 203: How They Work

To compute the available extents needed for the largest segment in the tablespace metrics 003/203 collect data used in calculations as follows:

1. All non-temporary tablespaces are selected from `DBA_TABLESPACES` and the free holes from `DBA_FREE_SPACE` (coalescing the space for dictionary tablespaces) are cached into an internal structure `(free_space)`.
2. A check for data files with auto extend enabled is made:
   `(AUTOEXTENSIBLE = 'YES' from DBA_DATA_FILES)`.
   If enabled, for the file system where the data file resides the total available free space amount is calculated:
   `(total_fs_space (in BLOCKS))`
3. The maximum size allowed for the data file is calculated:
   `((MAXBLOCKS - BLOCKS)/INCREMENT_BY) * INCREMENT_BY from DBA_DATA_FILES)`
   as well as the maximum file system space available to autoextend `((total_fs_space/INCREMENT_BY) * INCREMENT_BY)`

   The lesser of the two resulting values is used as the amount of file system space available `(fs_free_space)` and this amount is added to the over all free space `(free_space)`.
4. A count of the number of extents that are able to fit in the "holes" `(free_space)` is then tallied:
   `(max(NEXT_EXTENT) from DBA_SEGMENTS)`
   If the count value is less than or equal to the command line threshold, the value is sent to the OVO agent.

Metric 003: The Thresholds

Metric 003 has a maximum monitor template threshold of 0.5 to force an alarm on one or more tablespaces that cannot extend. You can find the monitor template for this metric in DBSPI-Oracle→ DBSPI-Oracle: Standard Metrics→DBSPI-Oracle: Metrics. To use Metric 003, deploy the template DBSPI-0003 and add the metric number to a collector template program, running every 5, 15 or 60 minutes. For example, to run Metric 003 instead of metric 203, and to set the threshold at 5 extents, you would modify the collector template command line in the DBSPI-Ora-15min-Favorites to read:

```
dbspicao -c DBSPI-Ora-15min-Favorites -m
3:5,16:1,17:80,18:95,58,62,64-65,77-80
```

Quick Start Metric 203: How It Works

Metric 203 monitors *dictionary tablespaces* differently from *locally managed tablespaces*. For dictionary tablespaces, it returns the total number of extents in the largest segment before it cannot extend.

For *locally managed tablespaces*, it determines only if there is enough room for the number of extents in the metric parameter *plus one*. For example, if metric 203 is run with a command line metric parameter of 5 extents (set in the collector template), the following output could be returned:

```
dbspicao -m 203:5 -p -v -i ora901

Instance: 'ora901' @ '/opt/oracle/product/9.0.1'
------------------------------------------------------------

ora901 0203 6.00 objext=CWMLITE tablespace_name=CWMLITE extents=6

ora901 0203 6.00 objext=DRSYS tablespace_name=DRSYS extents=6
```

```
ora901 0203 6.00 objext=EXAMPLE tablespace_name=EXAMPLE extents=6

ora901 0203 6.00 objext=INDX tablespace_name=INDX extents=6

ora901 0203 6.00 objext=LOCAL_AUTO tablespace_name=LOCAL_AUTO extents=6

ora901 0203 6.00 objext=OEM_REPOSITORY tablespace_name=OEM_REPOSITORY
extents=6

ora901 0203 10989.00 objext=SYSTEM tablespace_name=SYSTEM extents=10989

ora901 0203 6.00 objext=TOOLS tablespace_name=TOOLS extents=6

ora901 0203 195979.00 objext=UNDOTBS tablespace_name=UNDOTBS extents=195979

ora901 0203 6.00 objext=USERS tablespace_name=USERS extents=6
```

For *dictionary tablespaces* (UNDOTBS & SYSTEM), it returns the total number of extents available for the largest segment (10,989 & 195,979). However, for performance reasons, the algorithm stops when it determines that locally managed tablespaces have at least <*n*>+1 extents available where <*n*> is the command line threshold. With a threshold set at 5 (5+1), you can see why a value of 6.00 is returned for each locally managed tablespace in the above example (CWMLITE, DRSYS, EXAMPLE, etc.).

Metric 203: The Thresholds

Metric 203 uses a minimum threshold since it alarms when a tablespace has fewer than <*n*> extents before running out of space. The default monitor template threshold is one 1, while the default (collector template) command line threshold is 10:

```
dbspicao -c DBSPI-Ora-15min-Favorites -m
203:10,16:1,17:80,18:95,58,62,64-65,77-80
```

In this example, for dictionary tablespaces the DB-SPI collector would return the total number of extents available. For locally managed tablespaces it would return up to 11. When a tablespace has only 1 or 0 extents available before it runs out of space, an alarm occurs.

To increase the default of 1 or 0 extents before an alarm occurs, you would need to modify the monitor template to a value greater than 1. However, if you were to change the monitor template to a threshold greater than 10, you would also need to change the collector template command line threshold as well. For example, if you wanted to alarm for 20 or fewer free extents, you would change the command line in DBSPI-Ora-15min-Favorites to:

```
dbspicao -c DBSPI-Ora-15min-Favorites -m
203:20,16:1,17:80,18:95,58,62,64-65,77-80
```

In addition, you would change the DBSPI-0203 monitor template threshold to 20 as well. The larger the number, the slower the metric runs as more calculations are required for locally managed tablespaces in determining how many extents are available. By setting the default (collector template) command line threshold to 10, you need to modify only the OVO template unless you change the threshold to greater than 10.

Metric 6 and Metric 206 - Percentage of Free Tablespace

Metrics 006/206 calculates the percentage of available space in a tablespace as follows:

1. It selects all non-temporary tablespaces from DBA_TABLESPACES, then calculates the amount of space available from DBA_FREE_SPACE as a sum(BLOCKS) from DBA_FREE_SPACE.

2 It next calculates the space used by the data files as `sum(BLOCKS)` from `DBA_DATA_FILES` for the tablespace.

3 If any of the data files have autoextend enabled (`AUTOEXTENSIBLE = 'YES' from DBA_DATA_FILES`),
The total amount of space available within the file system where the data file resides is calculated:
`(total_fs_space (in BLOCKS))`

4 Now the maximum size allowed can be calculated for the data file (`(MAXBLOCKS - BLOCKS)/INCREMENT_BY) * INCREMENT_BY` from `DBA_DATA_FILES` and the maximum file system space available to autoextend (`(total_fs_space/INCREMENT_BY) * INCREMENT_BY`).
The lesser of the two values is used as the amount of file system space available: (`fs_free_space`).

5 The ratio is then calculated as:
`Ratio=(sum_dbafree_space + fs_free_space)/(sum_dba_datafiles + fs_free_space) *100`

For metric 6, if the ratio is less than the command line threshold - count the tablespace in the alarm count. If no command line threshold is specified for 6, an internal threshold of 80 percent is used. For 206, the percentage is always sent to the agent to compare against the threshold.

Metric 6/206: The Thresholds

Metric 6 uses a minimum command line threshold, which is the percentage of free space in the tablespace. It counts the number of tablespaces that have less than the free space specified in the command line threshold and sends this value to the OVO agent. The threshold for DBSPI-0006 should always be set to.5 so that an alarm would occur for even 1 tablespace with low free space.

Metric 206, as a drill-down metric, has only one threshold setting. This setting is compared to the free space percentage for every tablespace. The threshold for monitor template DBSPI-0206 should be set to the appropriate percentage value (the default is 10%).

Not used: Because these metrics are not part of the Quick Start group, to use them, you would have to do the following:

1 Add the metric number to one of the collector templates gathering data every 5, 15, or 1 hour. (For metric 6, you will also need to set a collector template command line threshold).

2 Deploy the monitor template, which you can find in DBSPI-0006 and DBSPI-0206 in the DBSPI-Oracle→ DBSPI-Oracle: Standard Metrics →DBSPI-Oracle: Metrics template group.

Quick Start Metric 16 and Metric 216: Segments Unable to Extend

Metrics 16/216 now alarm if fewer than <*n*> extents are available for segments within a tablespace where <*n*> can now be specified as a command line threshold value. Metrics 16/216 also alarm when the segment's `EXTENTS` plus the threshold value exceeds the segment's `MAX_EXTENTS` value.

Metrics 16/216 are similar to metrics 3/203 in that they select all non-temporary tablespaces, then calculate the amount of free space from `DBA_FREE_SPACE` and the amount of space available from the files system the same way that metric 3/203 does.

Metrics16/216 then check for any segments that cannot extend because `EXTENTS` + <*threshold*> > `MAX_EXTENTS` (from `DBA_SEGMENTS`).

It then computes the number of extents for the largest segment (`max(NEXT_EXTENT)` from `DBA_SEGMENTS`) that will fit in the "holes" (free_space) and compares this calculated value against the command line threshold. If the count is less than the threshold, it then examines each segment's `NEXT_EXTENT` value to compute the number of extents available to it. It compares this count against the command line threshold and sends the value to the OVO agent if the count is less than the command line threshold.

Quick Start use: Since metric 216 can be fairly slow, with the possibility of tens of thousands of segments, it is not included in Quick Start while metric 16 is. The default command line threshold is 1, which says to count the number of segments with 1 or 0 extents available.

Metric 16: The Thresholds

For metric 16, you will need to change the command line threshold that is specified on the command line in the template DBSPI-Ora-15min. To set the threshold to 5 extents, for example, change the command line like this:

```
dbspicao -c DBSPI-Ora-15min -m 203:10,16:5,17:80,18:95,58,62-65,77-81"
```

Leave the DBSPI-0016 templates at the threshold of .5 since this value represents the count of segments and for even 1 segment, an alarm is desired.

Metric 216: The Thresholds

For metric 216, the number of extents before sending an alarm must be passed on the command line threshold. For example, if you wanted to use 216 instead of 16 with a threshold of 3 extents, then you would change the command line like this:

```
dbspicao -c DBSPI-Ora-15min -m 203:10,216:3,17:80,18:95,58,62-65,77-81"
```

The above setting of "3" for metric 216 tells DB-SPI to send a value to the OVO agent for all segments that have 3 or fewer extents available. If no command line threshold is specified, the default is 1.

You do not need to change monitor template DBSPI-0216 because, by default, its threshold is set to 100. This maximum value setting is required because the "true" threshold is set on the command line of the collector template. Whenever DB-SPI makes a call to the OVO agent with the number of extents left in the segment, the 100 setting ensures that an alarm is generated. Changing the threshold in monitor template DBSPI-0216 would be necessary only if the command line threshold were set to a value higher than 100, which is highly unlikely.

Oracle SQL Query Monitor Templates

Within the Oracle *Add-Ons→Additional Metrics* subgroup are templates covering SQL query data. The templates are available in both roll-up (numbered from 101-108), and drill-down (301-308) modes. The templates monitor SQL queries as follows:

- Template DBSPI-0101 and DBSPI-0301: SQL disk reads
- Template DBSPI-0102 and DBSPI-0302: SQL fetches
- Template DBSPI-0103 and DBSPI-0303: SQL row scans, table scans
- Template DBSPI-0104 and DBSPI-0304: SQL execution rates
- Template DBSPI-0105 and DBSPI-0305: SQL buffer gets
- Template DBSPI-0106 and DBSPI-0306: SQL elapsed time
- Template DBSPI-0107 and DBSPI-0307: SQL CPU maximum time
- Template DBSPI-0108 and DBSPI-0308: SQL full table scan

How SQL Query Data is Collected

The metrics corresponding to the above templates, except DBSPI-0108 and DBSPI-0308, are dependent on the metric E100-SQLDataGatherer. The E100-SQLDataGatherer metric collects SQL query data by executing certain SQL statements to retrieve and store data in a database specific “pstore” (Persistent Store).

How SQL Query Data is Analyzed

After you deploy SQL query monitoring templates, DBSPI-0100 collects *all the data*. A specific area of the collected data is targeted for analysis when any one of the above listed templates, except DBSPI-0108 and DBSPI-0308, is deployed on a node.

Roll-up and drill-down templates both have thresholds set within them and generate alarms as appropriate to their individual thresholds. For both roll-up and drill-down templates, whenever an alarm is triggered, you can view the 10 top values in an automatic report, located in the generated message’s *Annotations*, which by default is set to 10. Please note that “10” means the highest 10 values, which includes at least one offender, but could include as many as 9 others that are not offenders. If only one violation occurs, you will still see the top 10 values in the generated message’s Annotations.

At the console, see the Annotations tabbed page by double-clicking the generated message.

Roll-up and Drill-down Template Differences in SQL Query Monitoring

The difference between roll-up and drill-down templates is in how they generate alarms on the data:

Roll-up templates trigger an alarm on the top query, regardless of whether there is one or multiple threshold violations. When one violation occurs, a message is sent, containing the top value and its owner (sql_id). In addition, an automatic action occurs, generating a report in the Annotations area of the message details. This automatic action report shows the offender and the top 9 values for the specific SQL query type (unless modified, the number shown is always 10).

Drill-down templates trigger alarms on the top 10 queries. When any violations occur, messages are generated for up to ten offenders. Like the roll-up template, the drill-down template also generates an automatic action report, viewable in the Annotations area of the message details. The report lists the worst offender and 9 others with the highest values to equal the default top 10 count (the count set in the Command line of the collector template, (see Note below).

> By default, the SQL query monitoring metrics are included in collector template **Ora-Add-Ons-15min**. At the console, you can double-click the Ora-Add-Ons-15min template and view the count value set for each drill-down template in the Command line text box. It is suggested you make a copy of the collector template that you want to use for deployment of the SQL query monitoring metrics. Then copy their metric numbers into the command line, remembering that you MUST include DBSPI-0100 for any of the templates (101-107 and 301-307) to work.

Using Metric Reports Tools to View SQL Query Monitoring Data

You can manually generate SQL query monitoring metric reports, rather than view them only as automatic action reports, triggered from an alarm.

To manually display customized metric information in a simple report, follow these steps:

a. At the OVO console, open the **Tools→SPI for Databases→ DBSPI Oracle→Oracle Reports (UNIX)** or **(Windows)**

b. Right-click the tool associated with the metric and select **Properties**.

c. Click the **Details** tabbed page and in the Parameters text box, modify the syntax as follows:

Syntax:
`dbspicao -m <metric_number>:<count> -r 2`

Default example (which shows top 10):
`dbspicao -m 101:10 -r 2`

Modified example (to show top 20):
`dbspicao -m 101:20 -r 2`

> **Automatic Action Parameter** `-r 1` means run the report based on an alarm; this parameter is used to generate the report shown in the Annotations area of the message details.
>
> **Tool parameter** `-r 2` means re-collect the data and report; this parameter refreshes the data.

d. Close the Tool Properties page.

e. Double-click the tool and select the node on which you want to generate the report.

Template Summaries

Data collection occurs, by default, at 15-minute intervals. Both roll-up and drill-down metrics analyze the data accumulated during the period between the last and current run of the collector template (the Ora-Add-Ons-15min template). (Refer to the SPI for Databases *Oracle Reference Guide* for more information about the settings within these templates.)

DBSPI-0100 (E100_SQLDataGatherer

This metric is run to collect SQL query data. The collected data is analyzed by the roll-up and drill-down templates (described below and in the SPI for Databases online help). The metric gathers data from the v$sqlarea table.

DBSPI-0101 (E101_SQLDiskReadsMax) and DBSPI-0301 (E301_SQLDiskReads)

These metrics analyze data available through metric 0100 covering Disk_read(s) retrieved from the v$sqlarea. A violated threshold of 05.0 triggers a warning message and indicates that one or more SQL statements have a high number of disk reads per execution, possibly impacting I/O performance. The low threshold encourages customization as appropriate to the environment.

DBSPI-0102 (E102_SQLFetchesMax) and DBSPI-0302 (E302_SQLFetches):

These metrics calculate the number of fetches per metric execution for the top <*number*> SQL queries during the period between the last and current run of the collector schedule. When a threshold of 150 is reached, a warning message is sent to the OVO message browser, indicating that one or more SQL statements have a high number of fetches per execution, possibly affecting performance.

DBSPI-0103 (E103_SQLScanRowsMax) and DBSPI-0303 (E303_SQLScanRows):

These metrics calculate the number of rows scanned per execution for top <*number*> SQL queries during the period between the last and current run of the collector schedule. When a threshold of 5 is reached, a warning message is sent to the OVO message browser. The low threshold encourages customization as appropriate to the environment. An exceeded threshold can indicate that excessive row scans are occurring, possibly affecting the physical or logical I/O performance.

DBSPI-0104 (E104_SQLExecRateMax) and DBSPI-0304 (E304_SQLExecRate):

These metrics calculate the execution rate per minute for top <*number*> SQL queries during the period between the last and current run of the collector schedule. When a threshold of 5 is reached, a warning message is sent to the OVO message browser. The low threshold encourages customization as appropriate to the environment. The metric can indicate that one or more SQL statements have a high execution rate, possibly affecting performance.

DBSPI-0105 (E105_BufferGetsPerExecRatio) and DBSPI-0305 (E305_BufferGetsPerExecRatio):

These metrics calculate the number of logical reads (buffer gets) per execution for top N SQL queries during the period between last and current run of the collector schedule. An exceeded threshold means that one or more SQL statements have a high number of buffer gets per execution, which may mean inefficient SQL statements. Inefficient SQL statements can result in poor logical I/O performance.

DBSPI-0106 (E106_SQLElapsedTime) and DBSPI-0306 (E306_SQLElapsedTime):

This metric looks for any SQL statement with a high elapsed time per execution. When one is encountered (threshold of 1.0 is reached), a warning message is sent to the OVO message browser. The low threshold encourages customization as appropriate to the environment. The metric can indicate that one or more SQL statements have a high execution rate, possibly affecting performance.

DBSPI-0107 (E107_SQLCPUTimeMax) and DBSPI-0307 (E307_SQLCPUTimeMax):

These metrics monitor all SQL statements for high CPU time per execution during the period between the last and current run of the collector schedule. When an SQL statement is encountered with high CPU time, a warning message is sent to the OVO message browser. A low default threshold encourages customization as appropriate to the environment. The metric can indicate one or more SQL statements have a high CPU time, possibly affecting logical I/O performance.

DBSPI-0108 (E108_SQLFullTableScanMax) and DBSPI-0308 (E308_SQLFullTableScanMax):

DBSPI-0108/0308 is a stand-alone template and has no dependency on DBSPI-0100 for its data collection.

These metrics monitor SQL statements performing full table scans during the period between the last and current run of the collector schedule. When an SQL statement is encountered that is performing a full table scan, a warning message is sent to the browser. One or more SQL statements performing a full table scan can result in queries running more slowly than expected.

Checking the Oracle Listener

The Database SPI allows you to perform two different types of checks on the Oracle listener, using two different templates. The first, the basic listener check, determines whether or not the default listener is running or you can configure it to check one or more specified listeners. The second, the advanced listener check, determines whether or not the listener can connect to a specific database instance. The two templates checking listener activities are:

- **DBSPI-Ora-Listener** (UNIX only*)* performs the *basic check*, which verifies that a default listener or a specified listener/listeners are running on the managed node (where the database server resides).
- **DBSPI-Ora-Listener Connect** (UNIX & NT) performs the *advanced check*, which verifies that the listener on the managed node can connect to the specified database instance. (The check is done by simulating the connection on the managed node.)

The Oracle listener check does not work with remote database monitoring, as described in Monitoring Remote Databases on page 151.

Using the Basic Listener Check

You can use the basic listener check template to look for the existence of any listener process (tnslsnr) or to check on a specific listener process. When used to check on any listener processes, the template checks for the absence of TNS errors when running the Listener Control Utility. If at least one *tnslsnr* process is found, everything is assumed to be all right. If any TNS errors are found, a problem is assumed, and an error is reported.

Specific listener processes can be checked as well, after you provide entries in the Database SPI configuration file. In such cases, the template can check multiple listeners by working in conjunction with the configuration file, where listener name(s) and, if necessary, password(s) are listed. See specific examples in Task #1 below.

This basic listener check is available only on UNIX.

Task 1: Configure Multiple Listeners (skip if not checking for specific listeners)

With no configuration the distributed *DBSPI-Ora-Listener* template monitors the default listener: `LISTENER`. To customize listener monitoring, edit the configuration file to cover UNIX nodes running listeners with non-default names, multiple listeners, and/or password-protected listeners. Then run the *Configure DB Connections* application against each node.

To enter listener information in the configuration file:

1. From the Window menu, select **Application Bank**.
2. In the Application Bank: window:
   for UNIX nodes select **DB-SPI→Admin** group, or
   for Windows nodes **DB-SPI→Admin Windows**.
3. Double-click **Configure DB Connections** to edit the file (see following page for syntax examples).

4 When finished editing the file, open the Node Bank window, navigate to the Database SPI nodes, and drag and drop each managed node onto the **Configure DB Connections** application.

Example 1: Monitoring multiple listeners and/or listeners with passwords.

```
SYNTAX_VERSION 4

 ORACLE
   HOME "<ORACLE_HOME>"
     DATABASE "<name>" CONNECT "<user/password>"
     ...
     DATABASE "<name>" CONNECT "<user/password>"
     ...
     LISTENER "<name>"
     ...

     LISTENER "<name>" CONNECT "password>"
```

Example 2: Monitoring listener(s) only, no database monitoring:

```
SYNTAX_VERSION 4

 ORACLE
   HOME "<ORACLE_HOME>"
     LISTENER "<name>"
     ...
```

The order in which you enter `LISTENER` and `DATABASE` information is not important.

Task 2: (required): Assign/distribute the basic listener check template.

1 Open the OVO Node Bank window and select the desired managed node(s) designated to run DB-SPI.

2 To assign (select) the templates for distribution from the Actions menu, select **Agents→Assign Templates**.

3 In the Define Configuration window click the **Add** button.

4 In the Add Configuration window click the **Open Templates Window** button.

5 In the Message Source Templates window select **SPI for Databases→DBSPI-Oracle→DBSPI-Oracle:Listener→ DBSPI-Ora-Listener**.

6 To distribute the assigned templates, from the Actions menu select **Agents→Install/ Update SW & Config**.

7 In the Install/Update OVO Software and Configuration window check the **Templates** check box.

8 Select the **Nodes in List Requiring Update** option button.

9 If you did not previously select a node, click the **Get Map Selections** button to list the target node(s) and click **OK** to distribute templates to the managed node(s).

Turning the Listen Check Off

You can disable the Oracle listener check running on a specific node by inserting an entry in the `defaults` file as follows:

1. At the managed node, open the file which is located according to operating system in:

   HP-UX, Linux, or Solaris: `/var/opt/OV/dbspi/defaults`
   AIX (DCE): `/var/lpp/OV/dbspi/defaults`
   AIX (HTTPS): `/var/opt/OV/dbspi/defaults`
   NT: `\usr\OV\dbspis\defaults`

2. Insert the following:
   `LISTENER_CHECK OFF`

3. Save, then close the file.

   To enable the listener check, remove the entry and save the file again.

Using the Advanced Listener Check

The *DBSPI- Ora-Listener-Connect* template checks Oracle listener *connections*. The advantage this template offers over the basic listener check is that you can verify not only that the listener is running but that it can successfully connect to the database. Unlike the *basic check*, though, the *advanced check* does require some configuration.

For connections between Oracle servers and clients, any of several mechanisms can be used in resolving an Oracle service name to a network address during a connection attempt. The possibilities include host naming (such as DNS or NIS in a TCP/IP environment), Oracle Names Server (an Oracle server that provides a directory service), or local naming (`tnsnames.ora`).

Two Possible Approaches to Checking Listener Connection(s)

The listener connection template relies on the last mechanism, the `tnsnames.ora` file to check the listener connection. You can choose between two approaches in determining how you want to use the template.

The DB-SPI default (approach #1): The DBSPI-Ora-Listener-Connect template includes a collection parameter (`-n` in the Command text box of the template) that directs the DB-SPI to map a service named in the Oracle `tnsnames.ora` file to an entry in the DB-SPI configuration file. The default setting, however, has limitations to it that are discussed in the following section.

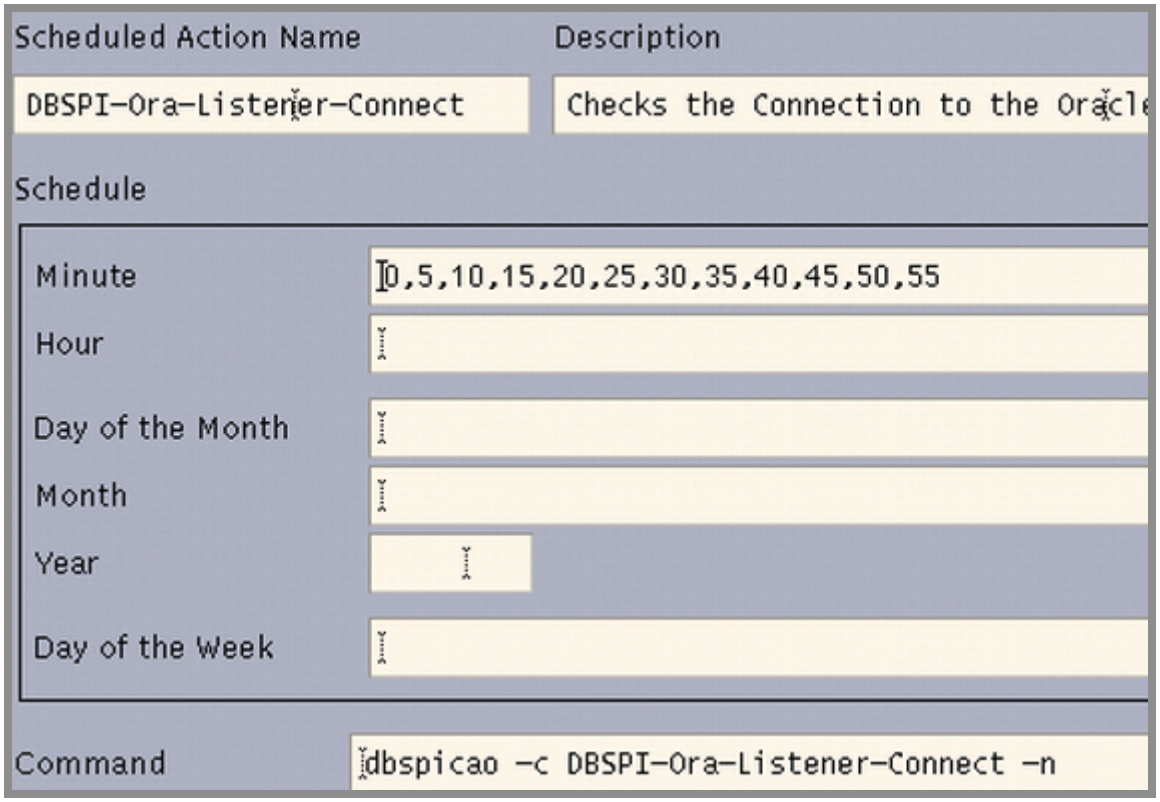
An alternative approach (approach #2): The DBSPI-Ora-Listener-Connect template can be modified to enhance its flexibility. Changing the template collection parameter allows for more accurate monitoring. This approach, however, requires not just modifying the template but inserting additional mapping entries in the DB-SPI configuration file.

To determine which approach to use, please review each of the following sections, referred to as Approach #1 or Approach #2.

Approach #1: (default) Checking for Nonspecific Connections

The Ora-Listener-Connect template contains a setting in its Command text box that directs the collector/analyzer (the `dbspicao` command) to perform a connection check. This setting is the "-n." option.

Use of the “-n” option requires no change to the template, only a mapping of the service name in the local `tnsnames.ora` file to an entry in the DB-SPI configuration file. Then, when the defined listener service connection fails, an error message is sent to the OVO message browser.



Prerequisites (-n option)

- You must have configured the DBSPI on the node.
- For each listener you want to check, you must configure your Oracle naming method (usually `tnsnames.ora`) with a service name that maps *exactly* to the instance name configured in the DB-SPI configuration file.

Results/Limitations

- If a connection using the listener is unsuccessful, a critical message is sent to the management server browser:
- If the connection using the listener is successful, no results are returned.

To map DB-SPI configuration file to the tnsnames.ora file:

1. From the Window menu select **Node Bank** and highlight the node on which you want to check the listener connection.
2. From the Window menu select **Application Bank**.
3. In the Application Bank window double-click **DB-SPI→Admin**.
4. In the Admin window double-click **Configure DB Connections**.

DB-SPI and Oracle File Entries

When you open the DB-SPI configuration file for the first time, you see the file syntax. In the DB-SPI configuration file syntax shown below, the third line contains the placeholder for the database instance <*name*> entry. The actual name that replaces this entry is the name you would enter in the `tnsnames.ora` file as a listener service.

```
ORACLE
  HOME "<ORACLE_HOME>"
  DATABASE <name> CONNECT "<user/password>"
```

DB-SPI configuration file.

Entries for a single Oracle database:

```
ORACLE
  HOME "/opt/oracle/product/8.0.5"
  DATABASE ora805 CONNECT "dbspi/dbspipass"
```

Oracle file (`tnsnames.ora`)

Entries would be similar to:

```
ora805=
  (DESCRIPTION=
    (ADDRESS_LIST=
       (ADDRESS= (PROTOCOL=IPC)(KEY=ora805))
       (ADDRESS= (PROTOCOL=TCP)(HOST=ellie)(PORT=1521))
    )
    (CONNECT_DATA=(SID=ora805))
  )
```

The instance name *must* match the `tnsnames.ora` file service name *exactly*.

Approach #2: (customized) Checking for Specific Connections

The advanced listener check allows you to check specific listener service connections. In addition, the required mapping does not include having to enter a special database instance name as a service name in the `tnsnames.ora` file. However, this approach does require template customization and extra entries in the DB-SPI Configuration file.

Normally the DB-SPI configuration file has one entry for each database instance. For example, the sample DB-SPI configuration file entry below has just one Oracle database named `ora805`:

```
ORACLE
  HOME "/opt/oracle/product/8.0.5"
  DATABASE ora805 CONNECT "dbspi/dbspipass"
```

The above entry is required for the DB-SPI collector/analyzer to process metrics.

Entries for mapping connection aliases in DB-SPI to Oracle Files

Using the "-m 1" option requires adding connection alias entries (to each database instance name) in the DB-SPI configuration file. The connection alias tests connections to the specified Oracle listener service. As such, the connection alias name must match a service name in the tnsnames.ora file. Please see the example below.

DB-SPI Configuration File Entries

The sample DB-SPI configuration file below has two entries, one for metric processing of instance ora805, and one to test the listener connection to instance ora805 using service name ora805_L1:

```
ORACLE

  HOME "/opt/oracle/product/8.0.5"
  DATABASE ora805 CONNECT "dbspi/dbspipass"
  DATABASE listener_check1 CONNECT "dbspi/dbspipass@ora805_L1"
```

Oracle tnsname.ora File Entries

The local tnsnames.ora file would contain entries like:

```
ora805_L1=
  (DESCRIPTION=
    (ADDRESS_LIST=
       (ADDRESS= (PROTOCOL=IPC)(KEY=ora805))
       (ADDRESS= (PROTOCOL=TCP)(HOST=ellie)(PORT=1521))
    )
    (CONNECT_DATA=(SID=ora805))
  )
```

Customizing the Template

To customize the template, use both the -m and -i options to run an instance status metric (metric #1) against a listener service acting (for purposes of the check) like an instance. In addition, you should modify any metric processing templates you use on the managed node to exclude the special listener service "instances" using -e.

Prerequisites (-m option):

— You must have configured the DB-SPI on the node.

— For each Listener you want to check, you must have configured your oracle naming method (usually tnsnames.ora) with a service name that maps to an entry in the DBSPI Configuration file.

Potential results (-m option):

— If a connection is made to the listener, no messages are generated.

— If a connection cannot be made to the listener, a critical message is sent to the management server browser.

To modify the template:

1 From the Window menu select Message Source Templates.

2 In the Message Source Templates window select **SPI for Databases→DBSPI-Oracle→DBSPI-Oracle:Listener→ DBSPI-Ora-Listener-Connect**

3 Copy the template and change the name.
(for purposes of this example, we will rename the template **DBSPI-My-Ora-Listener-Connect**.)

4 Within the new template `Command` text box, modify the entries to include the "`-m 1`," "-c" followed by the new template name, and "`-i`" followed by the listener connection aliases to resemble:
`dbspicao -m 1 -c DBSPI-My-Ora-Listener-Connect -i listener_check1 -i listener_check2`

To avoid extra collection cycles, update all the other templates (such as DBSPI-Ora-05min, DBSPI-Ora-15min, DBSPI-Ora-1h, and DBSPI-Ora-1d) to exclude these special listener "instances." For example:
`dbspicao -c DBSPI-Ora-1d -m 42:20,56,61 -e listener_check1 -e listener_check2`

Scenario Examples: Advanced Listener Connect Check

Following are scenarios that show you various situations where the "`-m 1`" option is used.

Scenario #1: One Listener to One Instance

In this scenario, a managed node has one Oracle instance named `DB1` and one listener service identified in the `tnsnames.ora` file as service name `DB1_L1`. You must have created a customized a template to use the "`-m 1`" option.

In this example we rename the custom template to `DBSPI-EXAMPLE-ORA-Listener-Connect`.

To modify the custom template, you would change the syntax in the `Command` text box to insert the following parameters:

```
dbspicao -m 1 -c DBSPI-EXAMPLE-ORA-Listener-Connect -i DB1_L1
```

Also, you would need to exclude (using the `-e` option) these special Listener "instances" from other collector templates used for metric processing.

```
DB-SPI configuration file

ORACLE

HOME "/opt/oracle/product/8.0.5"

   DATABASE db1 CONNECT "user/pass"

   DATABASE DB1_L1 CONNECT "user/pass@DB1_L1"
```

Oracle file (Oracle tnsnames.ora):

```
DATABASE db1 CONNECT "user/pass"

DATABASE DB1_L1 CONNECT "user/pass@DB1_L1"

DB1_L1=

 (DESCRIPTION=

      (ADDRESS=(PROTOCOL=IPC) (KEY=db1) )

      (CONNECT_DATA = (SID=db1))
```

```
)
```

Scenario #2: One Listener to Multiple Instances

In this scenario, a managed node has three Oracle instances named: `DB1`, `DB2`, and `DB3`. Instances `DB1` and `DB2` share the same `Oracle Home`. Each database has its own listener, identified in the `tnsnames.ora` file as `DB1_L1`, `DB2_L2`, and `DB3_L3`.

Prerequisite: You must have renamed a custom template to use the "`-m 1`" option. For purposes of this example, the template name is `DBSPI-EXAMPLE1-ORA-Listener-Connect`

To modify the custom template, you would change the syntax in the `Command` text box to insert the following parameters:

```
dbspicao -m 1 -c DBSPI-EXAMPLE1-ORA-Listener-Connect -i DB1_L1 -i DB2_L2
-i DB3_L3
```

As always, you would need to exclude these special Listener "instances" from other collector templates used for metric processing.

DBSPI configuration file:

```
ORACLE
HOME "/opt/oracle/product/8.0.5"
DATABASE db1 CONNECT "dbspi/pass"
DATABASE db2 CONNECT "dbspi/pass"
DATABASE DB1_L1 CONNECT "dbspi/pass@DB1_L1"
DATABASE DB2_L2 CONNECT "dbspi/pass@DB2_L2
HOME "/opt/oracle/product/8.0.6"
DATABASE db3 CONNECT "dbspi/pass"
DATABASE DB3_L3 CONNECT "dbspi/pass@DB3_L3
```

Oracle tnsnames.ora file

```
DB1_L1=
 (DESCRIPTION=
      (ADDRESS=(PROTOCOL=IPC) (KEY=db1) )
      (CONNECT_DATA = (SID=db1))
 )

DB2_L2=
 (DESCRIPTION=
      (ADDRESS=(PROTOCOL=IPC) (KEY=db2))
      (CONNECT_DATA = (SID=db2))
 )

DB3_L3=
```

```
(DESCRIPTION=
     (ADDRESS=(PROTOCOL=IPC)(KEY=db3)
     (CONNECT_DATA = (SID=db3))
)
```

Scenario #3: Multiple Oracle Listeners to one Database Instance

In this scenario, a managed node has two Oracle instances named `DB1` and `DB2`. Instances `DB1` and `DB2` share the same `Oracle Home`.

One listener services the `DB1` database instance: `DB1_L1`.
Two listeners service the `DB2` database instance: `DB2_L2a` and `DB2_L2b`.

Prerequisite: You must have renamed a custom template to use the "`-m 1`" option. For purposes of this example, the template name is `DBSPI-EXAMPLE2-ORA-Listener-Connect`.

To modify the custom template, you would change the syntax in the `Command` text box to insert the following parameters:

```
dbspicao -m 1 -c DBSPI-EXAMPLE2-ORA-Listener-Connect -i DB1_L1 -i
DB2_L2a  -i DB2_L2b
```

As always, you would need to exclude (using the `-e` option) these special Listener "instances" from other collector templates used for metric processing.

DB-SPI configuration file:

```
ORACLE
HOME "/opt/oracle/product/8.0.5"
DATABASE db1 CONNECT "dbspi/pass"
DATABASE db2 CONNECT "dbspi/pass"
DATABASE DB1_L1 CONNECT "dbspi/pass@DB1_L1"
DATABASE DB2_L2a CONNECT "dbspi/pass@DB2_L2a"
DATABASE DB2_L2b CONNECT "dbspi/pass@DB2_L2b"
```

Oracle tnsnames.ora file:

```
DB1_L1=
 (DESCRIPTION=
      (ADDRESS=(PROTOCOL=IPC) (KEY=db1) )
      (CONNECT_DATA = (SID=db1))
 )

DB2_L2a=
 (DESCRIPTION=
      (ADDRESS=(PROTOCOL=IPC) (KEY=db2))
      (CONNECT_DATA = (SID=db2))
```

```
 )

DB2_L2b=
 (DESCRIPTION=
      (ADDRESS=(PROTOCOL=IPC)9KEY=db2)
      (CONNECT_DATA = (SID=db2))
```

6 User Defined Metrics

To create user-defined metrics you must use SQL code outside this product to access the database-generated metric data. Once the metric data is available, DB-SPI provides a number of features that specifically retrieve and interpret it. Those features and the sequence of tasks you must complete to capture the data for use with OVO are covered in this chapter.

The set-up tasks for all databases are the same, but because each database requires specific code and specific entries in the configuration files, database-specific examples are included in the section Examples on page 142. Use this chapter to understand the components required, both outside and inside DB-SPI, that you use to start monitoring your database in an entirely custom way.

Understanding UDM Components

User-defined metrics (UDMs) are a powerful means for adding your own metrics to those pre-configured for DB-SPI. A UDM is SQL code that you create, and then configure the DB-SPI to execute and interpret in OVO. DB-SPI supports UDMs for Oracle, Microsoft SQL Server, Informix, and Sybase databases.

Each database type requires that you have knowledge of code/utilities for purposes as outlined in the following table:

Table 32 Database type code/utilities to use to create UDMs

Database	Metric COLLECT utility	Metric REPORT utility
Oracle	PL/SQL	SQL*Plus
MS SQL Server	Transact-SQL	ISQL or OSQL
Informix	Informix SQL.	dbaccess
Sybase	Transact-SQL	isql

DB-SPI fully supports user-defined metrics (UDMs) by including features that help you configure, retrieve, and display metric data that you define.

To create UDMs and set them up, use DB-SPI components as follows:

1. *Configuration file* (the dbspi`udm.cfg` file to configure UDMs): This file is available through the `Configure UDM` application. In the configuration file, you set up the user-defined metric numbering with the code (or you invoke the stored procedures or reference the `INCLUDE` file(s) containing code) for retrieving the metric data.
2. *Templates/features:* DB-SPI provides templates and features that handle UDM data as follows:

- *monitor* templates provide a means for setting up an alarm condition and triggering messages and reports
- *scheduled task* templates establish how often to execute data collection
- *reports* associated with the defined metric that can run as a tool or as an *automatic* or *operator action*
- *graphs* that allow you to graph UDM metric values.

To define metrics, you can make use of the following applications available in the `DBSPI Admin` application group:

- `Configure UDM` to create the UDM configuration file.

  For Oracle and MS SQL, which run on Windows systems, you select the Windows managed node, and open the `DBSPI→Admin Windows` group. Within this group you have access to the `Configure UDM` application for Windows.

- `Create SP/Oracle UDM`, `Create SP/MSSQL UDM`, `Create SP/Informix UDM`, or `Create SP/Sybase UDM` to define stored procedures for creating UDMs, for Oracle, MS SQL, Informix, and Sybase respectively.

UDM configuration file

The DB-SPI user-defined metric (UDM) configuration file (`dbspiudm.cfg`) lets you assign a number to each metric and specify its data source(s). Within the configuration file are beginning section keywords and required entries. Entries following the database type identify the metric to be defined, call stored procedures or code that creates the metric, and reference the `INCLUDE` file(s) containing SQL statements (that retrieve metric data or metric reports).

Specifically the structure of each definition is as follows

- The top line identifies the database type (such as `ORACLE`).
- The first line of the definition is the UDM number declared by the keyword `METRIC`.
- The second line is the metric component declared by the keyword `COLLECT`, followed by collection options and `SQL` code. The `COLLECT` component is required, and only one allowed per UDM.
- Subsequent lines can be used for one or more report components, each declared by the keyword `REPORT`.

When deployed to the managed nodes, the configuration file (and accompanying `INCLUDE` files if used) enables metric data to be retrieved and identified. Scheduled task and monitor templates (that you set up in later tasks) deal with sampling (collecting) and interpreting (alarming on) the retrieved data.

While stored procedures offer the potential advantage of better system performance, the disadvantage is that the procedure must be created and stored on each managed node and each database instance. Because of the tediousness of this configuration requirement, the recommended approach is that you insert SQL directly into the UDM configuration file.

UDM Configuration File Syntax

When you start Configure UDM, a file opens containing sample syntax as follows:

```
ORACLE
  METRIC 07XX
    COLLECT <OPTIONS> "<sqlcode>"
```

```
      REPORT 1 "<sqlcode>"
      REPORT 2 "<sqlcode>"
    METRIC 07YY
      ...
  INFORMIX
    METRIC 17XX
      COLLECT <OPTIONS> "<sqlcode>"
      REPORT 1 "<sqlcode>"
      REPORT 2 "<sqlcode>"
    METRIC 17YY
      ...
  SYBASE
    METRIC 27XX
      COLLECT <OPTIONS> "<sqlcode>"
      REPORT 1 "<sqlcode>"
      REPORT 2 "<sqlcode>"
    METRIC 27YY
      ...
  MSSQL
    METRIC 37XX
      COLLECT <OPTIONS> "<sqlcode>"
      REPORT 1 "<sqlcode>"
      REPORT 2 "<sqlcode>"
    METRIC 37YY

where <OPTIONS> are MW ITO RATE, e.g. COLLECT ITO MW RATE "<sqlcode>"
```

Table 33 UDM Keywords

Keyword	Description
ORACLE MSSQL INFORMIX SYBASE	Identifies the beginning of each section of the UDM configuration file specific to the named database type. The UDM configuration file can include one or more of these sections, depending on whether or not you are monitoring more than one database type.
METRIC	Each UDM metric definition must start with this keyword followed by a metric number. Oracle range: 0700 - 0799* MS SQL range: 3700 - 3799 Informix range: 1700 - 1799 Sybase range: 2700 - 2799
COLLECT	(Optional) Only one COLLECT statement is allowed per metric.
MW	(Optional) Sends value to HP OpenView Performance Agent (earlier called MeasureWare)
ITO	(Optional) Sends value to OVO for alarm generation.
RATE	(Optional) Allows rate calculations scaled to minutes using the formula: (<*current*>-<*previous*>)/<*seconds_between_collector_executions*>*60

Table 33 UDM Keywords

Keyword	Description
REPORT	(Optional) Multiple REPORT statements are allowed per metric.
<code_specification>	Specifies SQL to use for metric collection or report. Code specification is either a quoted string (must use double-quotes and use backslash [\] to escape embedded double-quotes) or an INCLUDE keyword followed by a file name in double-quotes (see details below).
INCLUDE	Statement with path name that contains the SQL statements for the COLLECT clause and statements for the REPORT clause. If the path is not absolute, the collector/analyzer tries to load it from the agent's monitor directory (HP-UX: /var/opt/OV/bin/OpC/monitor). In order to have OVO distribute the INCLUDE files to all managed nodes when you re-distribute monitors, you must place the files in the appropriate location on the management server (according to platform, see Table 34 below).

*The HP OpenView Smart Plug-In for PeopleSoft comes bundled with DB-SPI User-Definable Metrics (UDMs) in the number range 0790-0799. To avoid UDM number conflicts, if you use the Oracle Smart Plug-In and install the PeopleSoft Smart Plug-In, you must avoid using upper-range UDM numbers such as 0790-0799 and begin with lower-range UDM numbers such as 0700. Although reservation of UDM numbers will not be enforced by DB-SPI, adhering to this practice may help avoid the need to renumber UDMs and rename templates in the future if customer and Smart Plug-In UDMs overlap.

Any combination of the MW, ITO, and RATE keywords is legal; for example:

```
MW, ITO, RATE, MW ITO, MW RATE, ITO RATE, or MW ITO RATE.
```

Separate options with blanks. (A metric with no options or just the RATE option executes, but the data is not used, OVO does not alarm, and the data does not go to OV Performance Agent.) The only way to display the value from this type of configuration is by using the -p option on the command line.

Table 34 The INCLUDE Statement Paths

Managed Node OS	Location on Management Server
HP-UX	/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/s700/hp-ux10/monitor
Solaris	/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc/solaris/monitor
AIX	/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix/monitor
Windows	/var/opt/OV/share/databases/OpC/mgd_node/customer/ms/intel/nt/monitor

Putting It All Together

You create a UDM as follows:

- Determine the SQL Code to use
- Create and Distribute UDM Configuration File
- Test UDM
- Create and Distribute UDM Templates
- (Optional): Start the Enable UDM Graphs application

Setting Up the UDM Configuration File

To define metrics for UNIX or Windows systems, use your database-specific SQL code to direct the DB-SPI collector/analyzer to retrieve metric data from the database. Two methods exist for retrieving metric data: (1) call a stored procedure created on each managed node and called from the UDM configuration file or (2) use SQL statements either in the configuration file or in saved files that are referenced with `INCLUDE` entries in the configuration file. For more information, see Examples on page 142.

While stored procedures offer the potential advantage of better system performance, the disadvantage is that the procedure must be created and stored on each managed node and each database instance. Because of the tediousness of this configuration requirement, the recommended approach is that you insert SQL directly into the UDM configuration file.

Task 1: Determine the SQL Code to use

- If you are using a stored procedure, create the syntax to access it and insert the syntax into the configuration file (which you create in Task 2). See the section Examples on page 142 as a guideline.
- If you are using SQL statements, either place them in a DB-SPI `INCLUDE` file (which you can optionally create and reference in the configuration file), or insert the SQL statements directly into the configuration file (which you create in the Task 2).

  The alternative to the method below is to directly include the statements in the UDM configuration file policy that you create in Task 2

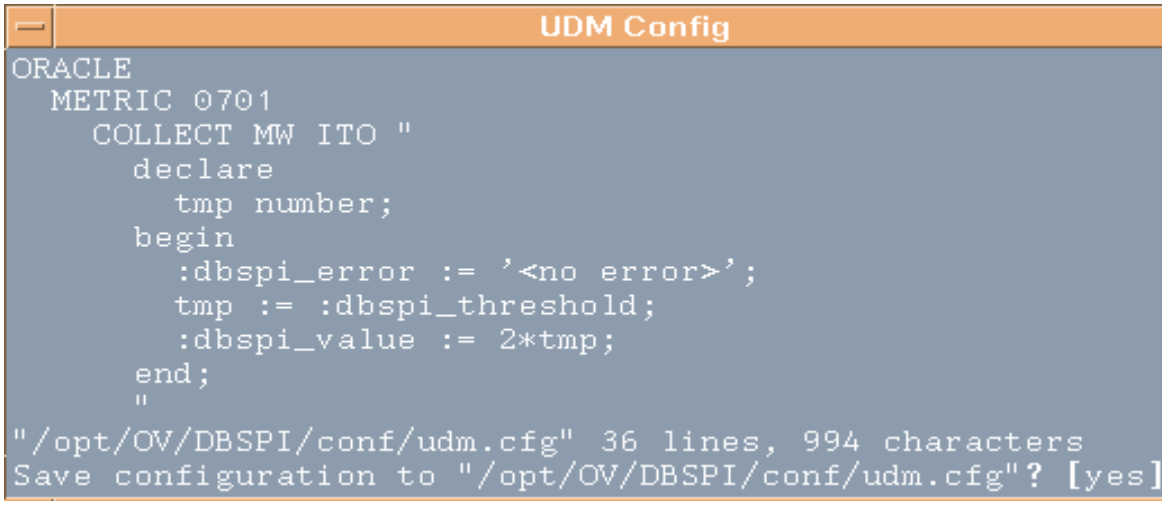
Task 2: Create and Distribute UDM Configuration File

Using the **Configure UDM** application, you can enter information about each UDM (user definable metric) and stored procedure (created optionally in Task 1). All UDMs, regardless of database type, are configured in one UDM file. Each database type has its own section in the file. If you are using OVO, you must distribute the UDM to the required nodes and activate it by distributing monitors and collectors at the end of the procedure.

1 In the DBSPI Admin or NT-DBSPI Admin application group start the **Configure UDM** application.

  The first time Configure UDM starts, it launches a text editor, showing a file with comments (#):

Figure 27 UDM configuration file with entries



2 Replace entries in the template with entries similar to the examples in Examples on page 142.

The metric code communicates with the collector/analyzer through the following host variables:

Table 35 Host Variables

Variable Name	Description
`:dbspi_error`	This output variable can be used to set an error text in case of an exception. The entered text will be added to the DBSPI error message if the *dbspi_value* is -1. Note, this variable must be the first host variable referenced in the PL/SQL block.
`:dbspi_threshold`	This input variable will be set with the command line threshold passed in with the -m option of the collector/ analyzer (for example, `-m 701:2` sets this variable to 2).
`:dbspi_value`	This output variable returns the metric value and must always be the last variable referenced in the PL/SQL block.

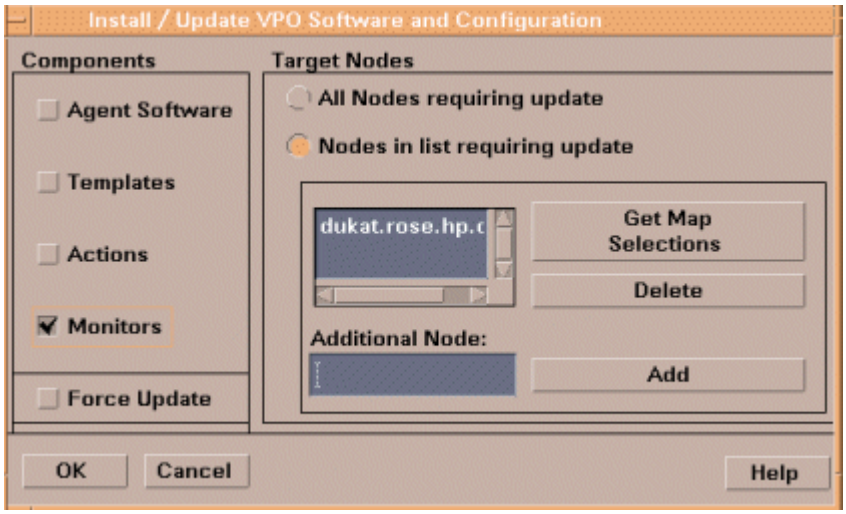
The variable for the `:dbspi_error` must always be first and the variable for `:dbspi_value` must always be the last as referenced in the PL/SQL block. If `:dbspi_threshold` is used, it must always be referenced between `:dbspi_error` and `:dbspi_value`, otherwise a runtime error occurs.

3 Save and exit. (A syntax check is performed when you exit the editor.)

4 If the syntax is correct, distribute the configuration to all nodes by selecting **Actions→Agents→Install/Update SW & Config**.

5 Click the **All Nodes, Monitors,** and **Force Update** option buttons.

Figure 28 Distribute the UDM Monitors to the desired Oracle Nodes



6 After successful distribution, verify the metric code by manually executing the collector/analyzer as described Task 3.

Task 3: Test UDM

1 Log on to the managed node as root (for UNIX) or administrator (for NT).

2 Add the path name for the <*OVO_monitor*> directory to the PATH environment variable.

```
For UNIX
ksh:

# export PATH=$PATH:<OVO_monitor_directory>
sh:

# PATH=$PATH:<OVO_monitor_directory>
# export PATH

For NT
set path=c:<OVO_monitor_directory>;%PATH%
```

3 Verify the metric with the following command (Oracle metric 700 is used as example. Use the command for the database-specific collector.):

```
dbspicao -p -m 700 -v
Instance: 'openview' @ '/opt/u01/home/oracle/product/7.3.4'
-------------------------------------------------------
openview 0700 0.00
```

For information about commands for other databases, see Table 63.

4 Execute the report for the metric (Oracle metric 700 is used as example. Use the command for the database-specific collector.):

```
dbspicao -r 1 -m 700 -v
Instance: 'openview' @ '/opt/u01/home/oracle/product/7.3.4'
-------------------------------------------------------
                    Report For Database openview
                      Mon Nov 16 14:01:54 1998
                    Metric UDM 0700 (Report 1)
```

```
Dirty Queue Length
------------------
                 4

Write Requests
--------------
          7130
```

For information about commands for other databases, see Table 63.

5 Do the same for 0701 but add the required command line threshold:

```
dbspicao -p -m 701:3 -v
Instance: 'openview' @ '/opt/u01/home/oracle/product/7.3.4'
------------------------------------------------------------
openview 0701 6.00
```

The above output indicates the correct execution of the code.

Task 4: Create and Distribute UDM Templates

DB-SPI provides the following template groups that can be used to create UDM templates:

`DBSPI-Oracle: UDM Templates`

`DBSPI-MSS7: UDM Templates`

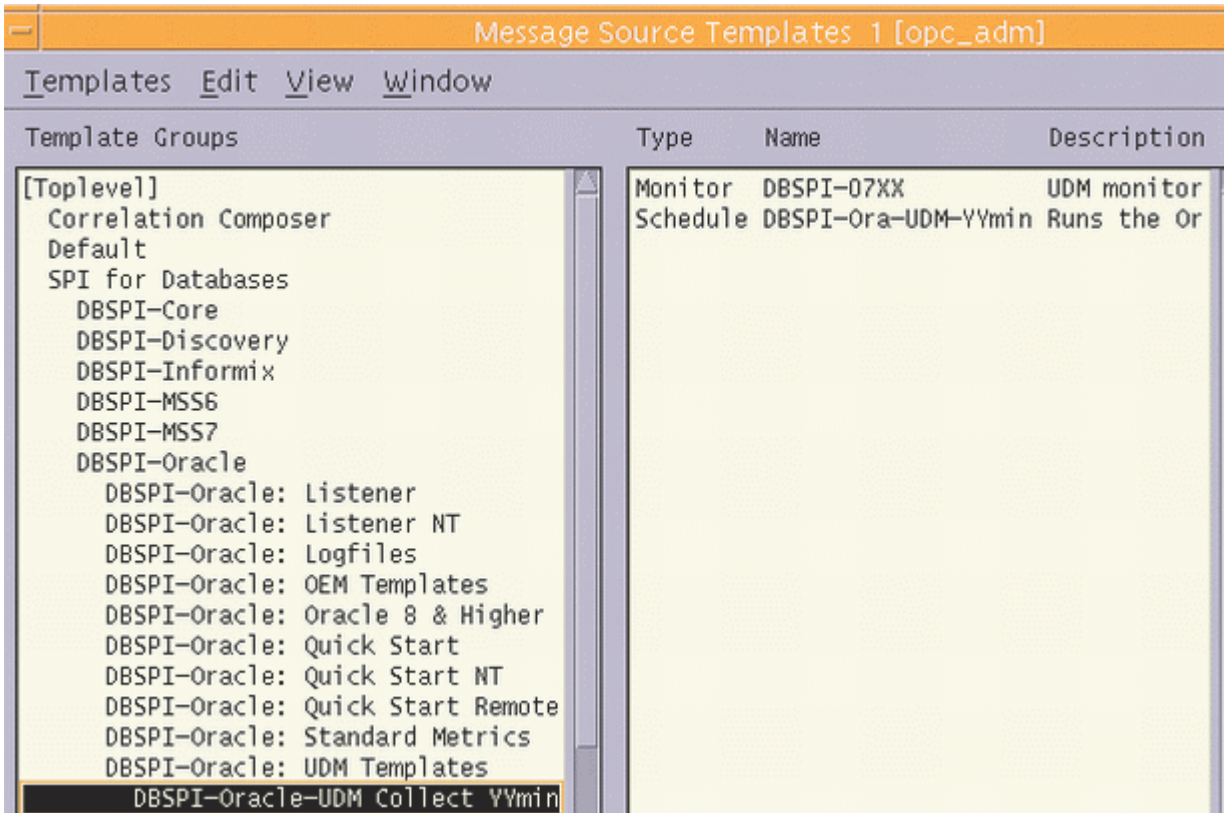
`DBSPI-MSS6: UDM Templates`

`DBSPI-Informix: UDM Templates`

`DBSPI-Sybase: UDM Templates`

The illustration below shows the templates `DBSPI-07XX` and `DBSPI-Ora-UDM-YYmin` that can be copied and modified for Oracle UDMs (do not use these templates as they are only examples).

Figure 29 Oracle UDM monitor templates

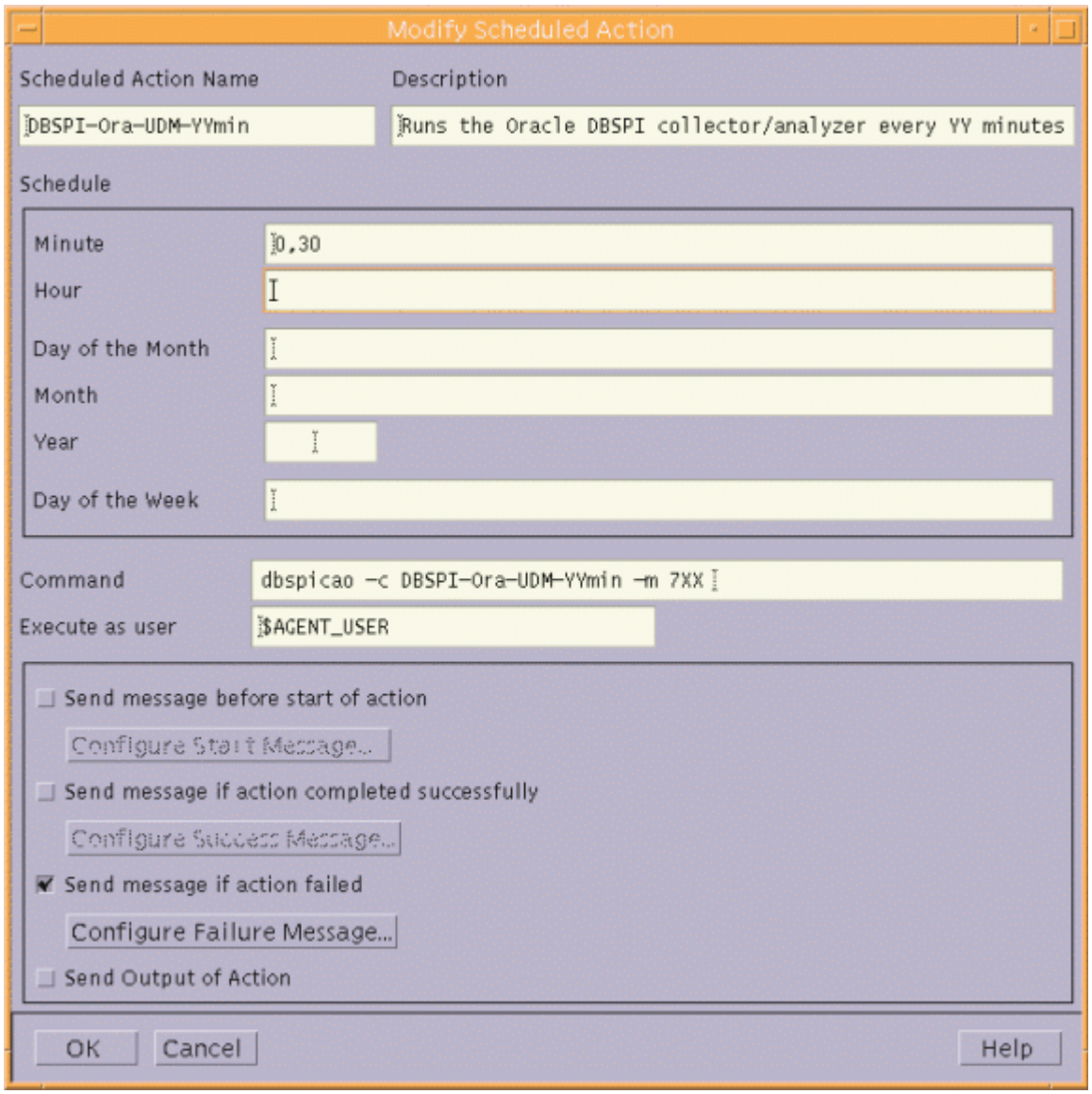


Two templates are necessary for every UDM: (1) the monitor template that defines the database metric conditions and (2) the collector template that schedules data collection for all metrics in the template group. The collector template tells the collector/analyzer to retrieve data at the specified collection interval for each listed metric in the template group.

In the following illustrations, **DBSPI-07XX** specifies the metrics which are collected for a specific interval, and **DBSPI-Ora-UDM-YYmin** schedules the collector/analyzer (dummy metrics are used in these examples).

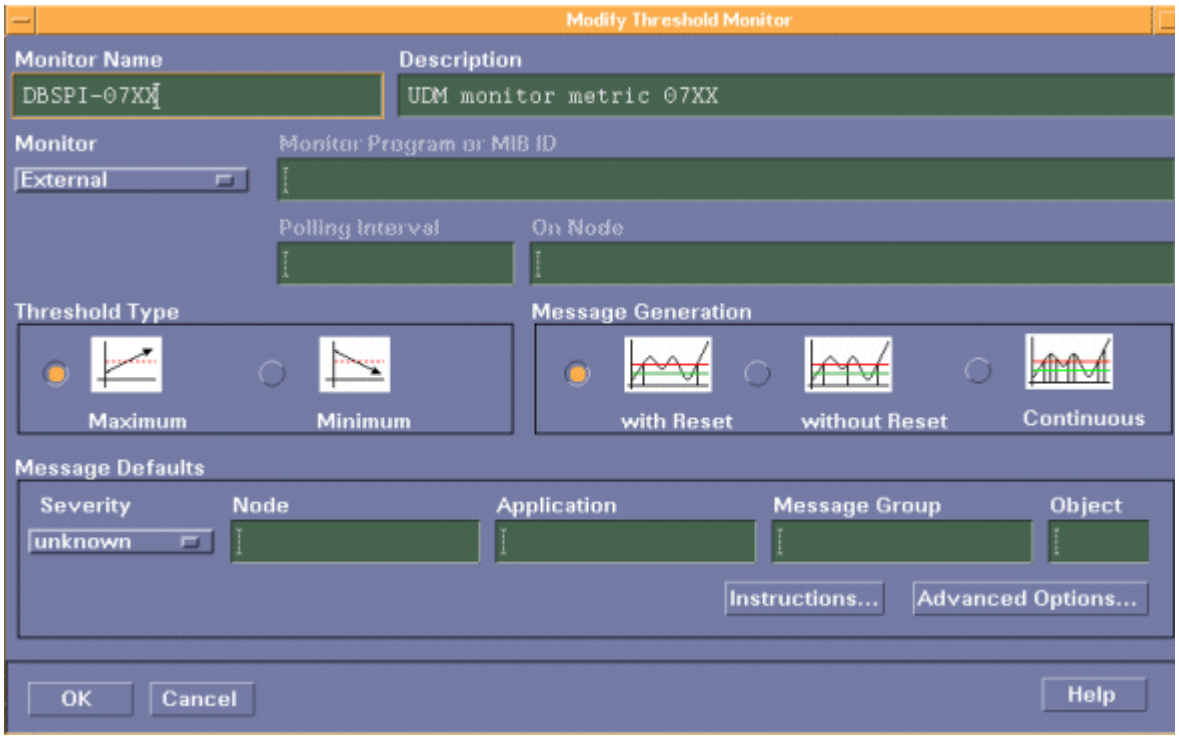
You could choose to add the UDM metrics to the standard collector templates. By doing this, you would improve performance slightly by avoiding an extra connection to the database.

Figure 30 In OVO, sample collector template DBSPI-Ora-UDM-YYmin



In the collector template, the **Command** text box contains the collector/analyzer command (**dbspicao, dbspicam, dbspicai, dbspicas**). The -c option must always correspond with the **Scheduled Action Name** box (in this case, `DBSPI-Ora-UDM-YYmin`). The `-m` option lists the metrics (7XX in this case) that are collected during this interval (set in the `Schedule` field). UDM metrics must fall within the range specified for each database type. See Table 33 on page 131 for more information.

Figure 31 In OVO, sample monitor template DBSPI-07XX



The Monitor field is always set to External for metric monitors.

Figure 32 In OVO, setting conditions for the UDM

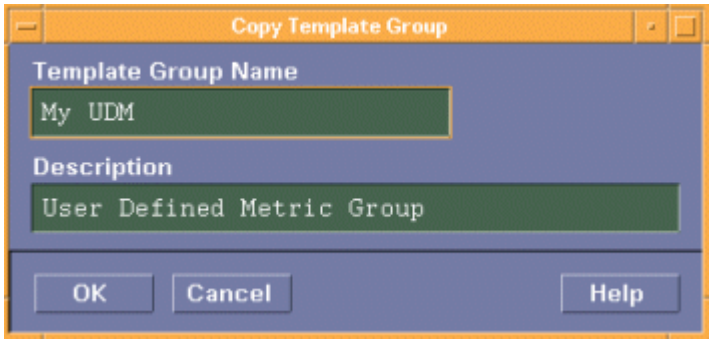
In this example, `DBSPI-07XX` implements the actual metric. The illustration above shows a dummy condition for monitor `DBSPI-07XX`. The naming of a condition adheres to the following convention: `monitor_name.condition_number` (see the Description field).

Example UDM Template Configuration

This section describes how an actual user-definable metric (UDM) is configured for Oracle. The sample UDM templates referenced below should not actually be changed. Make copies of these templates before making modifications.

1 Use the **Add...** button on Message Source Templates screen to add a new template group (for example, My Oracle UDM) under the group `DBSPI-Oracle`.

Figure 33 Creating a User-Defined Metrics Group



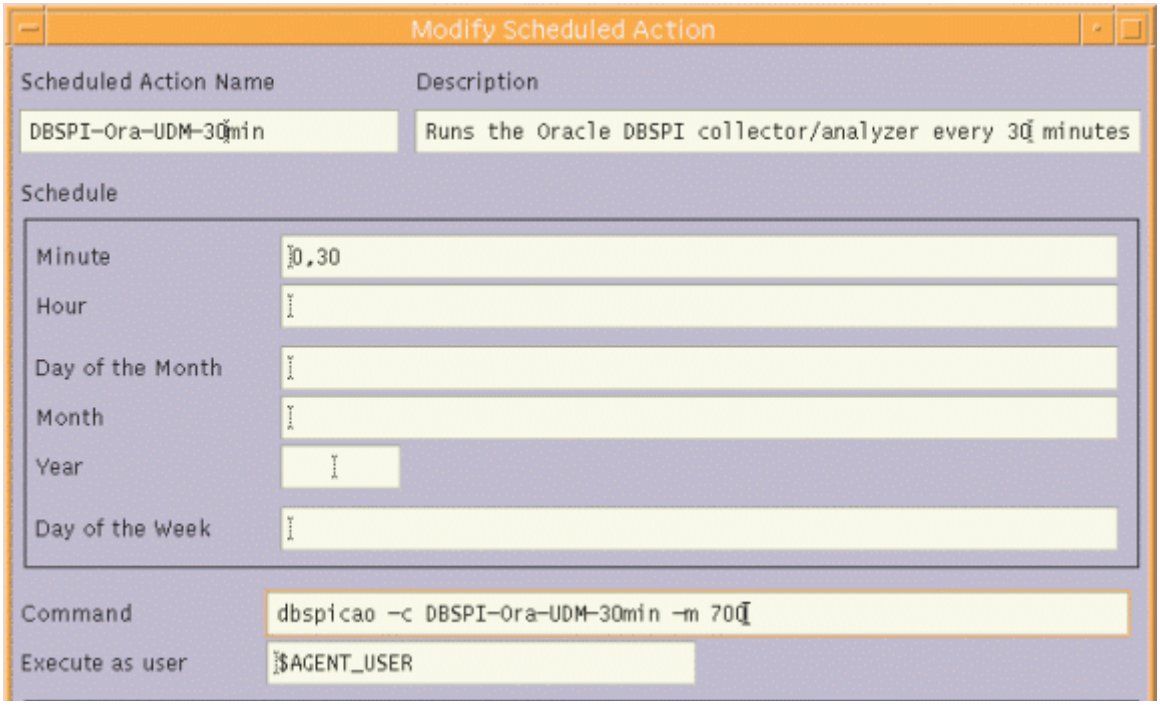
2 Select the sample templates `DBSPI-07XX` and `DBSPI-Ora-UDM-YYmin` from the `DBSPI-Oracle-UDM-YYmin` group and in OVO use **Edit→Copy** to copy the templates to the clipboard.

3 Select the new group (in this case, My Oracle UDM) and use **Edit→Paste** to paste the templates into the group.

4 To create the collector template, select **DBSPI-Ora-UDM-YYmin** from the new group, use the **Copy...** button to make a copy, and rename it, replacing "`YY`" with the number of minutes when the collector must be run (for example, 30).

5 In the collector template, in the **dbspicao** command line (**Command** text box) modify the collector name (after the -c option) to match the new name of the template, and change the "7XX" to a valid argument (for example, 700). The collector illustrated below will run `dbspicao -c DBSPI-Ora-UDM-30min -m 700` every 30 minutes.

Figure 34 Modifying the Sample Collector Template



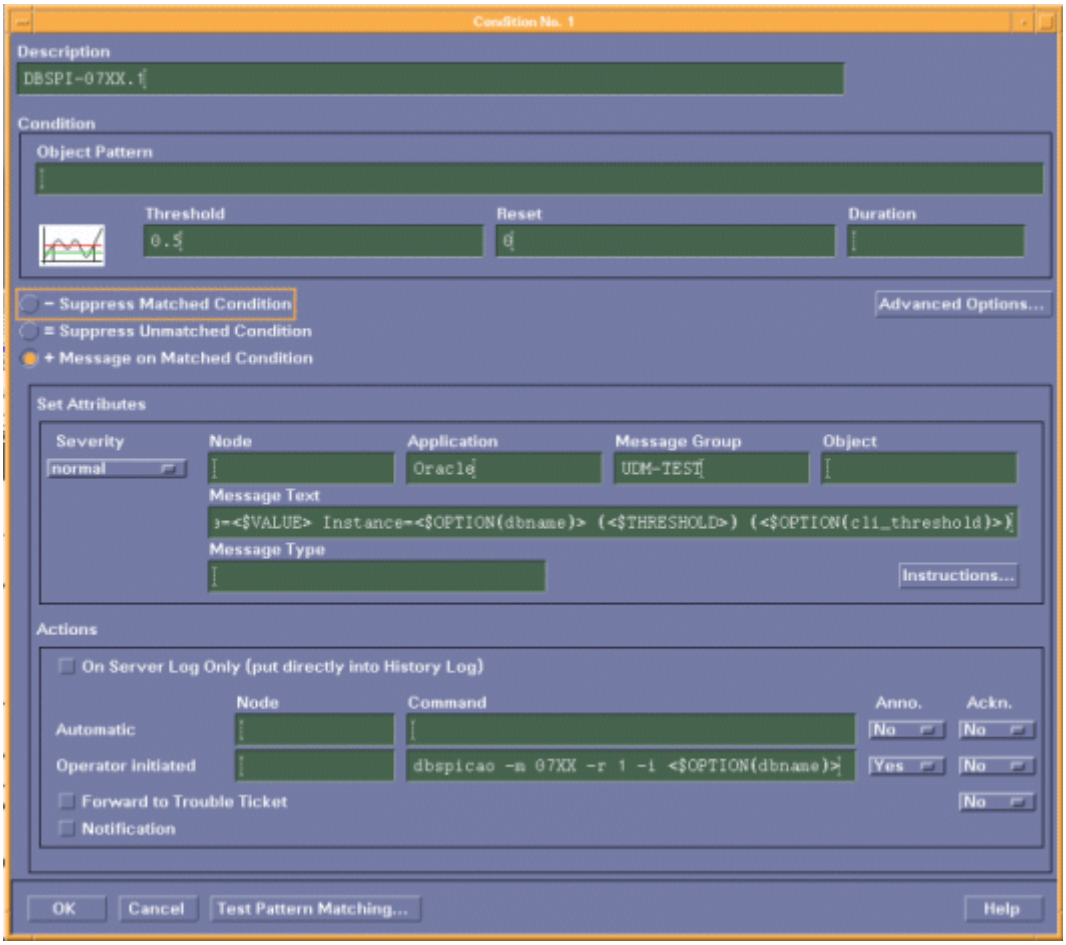
6 Select `DBSPI-07XX` and use the **Copy...** button to make a copy and rename it and replace "`XX`" with 00.

7 To create a condition for the new monitor, double-click **DBSPI-0700** and use **Conditions→Modify** to modify the default **DBSPI-07XX.1** condition.

8 Make changes similar to those shown in the illustration that follows. The Message Group could be changed to match the default DB-SPI Oracle message groups.

Multiple conditions with different severities and thresholds can be added to a single External monitor template.

Figure 35 Modifying the Monitor Condition



9 In the collector template **DBSPI-Ora-UDM-30min** add metric number 701:2 following the -m option. In this example, we pass the metric parameter 2 to `dbspicao` that will be used in the `:dbspi_metric_parameter` variable in the PL/SQL procedure.

10 Select **DBSPI-07XX** and use the **Copy...** button to make a copy and rename it to **DBSPI-0701**.

11 Use **Conditions→Modify** modify the default **DBSPI-07XX.1** condition.

In the `Set Attributes` section of the Condition screen, the following variables can be set:

Table 36 UDM Variables

Variable Name	Evaluates to
`<$VALUE>`	Actual monitor value that was generated by the metric
`<$THRESHOLD>`	Value specified in the "Threshold" field.
`<$OPTION (dbname)>`	ORACLE_SID of the database instance.
`<$OPTION (cli_threshold)>`	Threshold passed in on the command line.

The `Operator Initiated` field shows how a report execution is specified.

— The `-m` option must always reference the metric.

— The `-r` option indicates a report execution and is followed by the report number.

— The `-i` option must always be present and followed by `<$OPTION(dbname)>`.

12 After you have created all metrics, use the **Delete from Group** button to delete the sample templates `DBSPI-07XX` and `DBSPI-Ora-UDM-YYmin` from the new group.

Deleting XX and YY templates from the group prevents OVO errors from occurring. It is important that you NOT select **Delete From All** because this selection removes them from the UDM Templates group as well.

13 Assign and distribute the new template to required nodes by selecting **Actions→Agents→Install / Update SW & Config.**

Task 5: (Optional): Start the Enable UDM Graphs application

If you are running OV Performance Agent (earlier called MeasureWare Agent) on the managed node and would like to use metric data to create PerfView or HP OpenView Performance Manager graphs, complete the following steps:

1 Double-click the **DBSPI→Admin** (or **DBSPI→Admin Windows**) application group.

2 Enable the new template group by dragging the managed node to the **Enable UDM Graphs** icon in the DBSPI Admin applications group.

Examples

Oracle

Creating a Stored Procedure

Use the `Create SP/Oracle UDM` application to enter the stored procedure as follows:

```
create procedure dbspiudm0701 (dbspi_value out number) as
 block_gets number;
 consistent_gets number;
 physical_reads number;

begin
  select value into block_gets from v$sysstat
    where name = 'db block gets';
  select value into consistent_gets from v$sysstat
    where name = 'consistent gets';
  select value into physical_reads from v$sysstat
   where name = 'physical reads';
  dbspi_value := (((block_gets + consistent_gets) -
   physical_reads )/(block_gets + consistent_gets)) * 100;
end;
```

Sample Configuration File

```
ORACLE
```

 # Example for the use of stored procedures and INCLUDE statement:

```
METRIC 0701
COLLECT MW ITO "exec dbspiudm0701"
REPORT 1 INCLUDE "report_3701.sql"
```

 # Example for the direct use of code with the threshold passed from the collector command:

```
METRIC 0702
COLLECT MW ITO "
declare
sql1_val number;
sql2_val number;
tmp number;
begin
:dbspi_error := '<no error>';
tmp := :dbspi_threshold;

select value into sql1_val from v$sysstat where
name='sorts (disk)';

if sql1_val > tmp then

select value into sql2_val
from v$sysstat where name='sorts (memory)';

if sql2_val = 0 then
:dbspi_value := 0;
else
:dbspi_value := sql1_val/sql2_val;
end if;

else
:dbspi_value := 0;
```

```
end if;

exception
when OTHERS then :dbspi_value := -1 ;
end;
"

REPORT 1 "
select value \"sorts (disk)\" from v$sysstat where
name='sorts (disk)';
select value \"sorts (memory)\" from v$sysstat where
name = 'sorts (memory)';

  "
```

Microsoft SQL Server

Creating a Stored Procedure

Use the `Create SP/MSSQL UDM` application to enter the following stored procedure. This example shows a stored procedure with a user-configured default threshold of 20:

```
CREATE PROCEDURE sp_dbspiudm3702 @threshold FLOAT = 20.0
AS
DECLARE @dbspi_error VARCHAR(100)
DECLARE @dbspi_value FLOAT
set @dbspi_error='no_error'
select
   @dbspi_value=count(*)
from
   master..sysprocesses
where
   memusage >= @threshold
select @dbspi_error, @dbspi_value
RETURN
GO
```

Sample Configuration File

```
MSSQL
```

 # Example for the direct use of code:

```
   METRIC 3700
```

```
    COLLECT MW ITO "select '', 1.0 * sum(size) from master..sysfiles"
    REPORT 1 "select * from master..sysfiles"
```

 # Example for the use of stored procedures and INCLUDE statement:

```
# Before running this metric, install the stored procedure sp_dbspiudm3701
# using the app "Create SP /MSSQL UDM"

  METRIC 3701
    COLLECT MW ITO "exec sp_dbspiudm3701"
    REPORT 1 "select name,status from master..sysusers"
    REPORT 2 INCLUDE "report_3701.sql"



# This metric demonstrates use of threshold in a stored procedure.
# Before running this metric, install the stored procedure sp_dbspiudm3702
# using the app "Create SP /MSSQL UDM"
```

To define a threshold for a stored procedure, you must include a "?" after the name of
the procedure (see example below). The “?” is critical for passing in a threshold
parameter.

```
  METRIC 3702
    COLLECT MW ITO "exec sp_dbspiudm3702 ?"
```

The following example illustrates SQL in the UDM configuration file for **Microsoft SQL Server 2000.** This example shows a UDM with no user-configured threshold (that is, a threshold that is passed into the UDM from the command line, as in `dbspicam -m 3703:20`):

```
MSSQL
  METRIC 3703
  COLLECT ITO
  "EXEC(\"
DECLARE @dbspi_error VARCHAR(100)
DECLARE @dbspi_value FLOAT
set @dbspi_error='no_error'
select
@dbspi_value=count(*)
    from
master..syscacheobjects
   where
pagesused >= 10
select @dbspi_error, @dbspi_value
```

```
  \")"
```

The following example illustrates SQL in the UDM configuration file, for **Microsoft SQL Server 2005** instances:

```
MSSQL

  METRIC 3703

  COLLECT ITO

  "EXEC(\'

DECLARE @dbspi_error VARCHAR(100)
DECLARE @dbspi_value FLOAT
set @dbspi_error='no_error'
select
@dbspi_value=count(*)
    from
master..syscacheobjects
   where
pagesused >= 10
select @dbspi_error, @dbspi_value


  \")'
```

Informix

Creating a Stored Procedure

Use the `Create SP/Informix UDM` application to enter the following stored procedure:

```
create procedure dbspiudm1701 ()

   returning char(256), int;


define ret_value int;

define errmsg char(256);

define bufwrites int;

define diskwrites int;


let errmsg = 'No writes (zero divisor)';


SELECT value INTO bufwrites FROM sysmaster:sysprofile WHERE
NAME='bufwrites';

SELECT value INTO diskwrites FROM sysmaster:sysprofile WHERE
NAME='dskwrites';


if (bufwrites + diskwrites = 0)
```

```
then let ret_value = -1;
else let ret_value = ((bufwrites)/(bufwrites + diskwrites)) * 100;
end if;

return errmsg, ret_value;

end procedure;



create procedure dbspiudm1702 (threshold int)
   returning char(256), int;

define ret_value int;
define errmsg char(256);
define bufwrites int;
define diskwrites int;

let errmsg = '';

SELECT value INTO bufwrites FROM sysmaster:sysprofile WHERE
NAME='bufwrites';
SELECT value INTO diskwrites FROM sysmaster:sysprofile WHERE
NAME='dskwrites';

if (bufwrites + diskwrites < threshold)
then let ret_value = 100;
else let ret_value = ((bufwrites)/(bufwrites + diskwrites)) * 100;
end if;

return errmsg, ret_value;

end procedure;
```

Sample Configuration File

```
INFORMIX
```

 # Example for the direct use of code and use of the INCLUDE statement:

```
METRIC 1700
  COLLECT ITO MW "select '', count(*) from syssessions"
  REPORT 1 "select * from syssessions"
  REPORT 2 INCLUDE "inf700.sql"
```

 # Example for the use of stored procedure:

The “?” is critical for passing in a threshold parameter.

```
METRIC 1701
  COLLECT ITO MW "execute procedure sysutils:dbspiudm1701()"
METRIC 1702
  COLLECT ITO MW "execute procedure sysutils:dbspiudm1702(?)"
METRIC 1703
  COLLECT ITO MW "select '', count(*) from syschunks where nfree < ?"
```

Sybase

Creating a Stored Procedure

Use the `Create SP/Sybase UDM` application to enter the following stored procedure:

```
create procedure sp_UDM_dbspi2701
as
  select 'dbspiudm2701 error', 1.0 * sum(size) from master..sysusages
return
go


create procedure sp_UDM_dbspi2702 @threshold int = -1
as
  select 'sp_marks2 error', count(*) from master..sysusages
  where size > @threshold
return
go
grant execute on sp_UDM_dbspi2702 to public
go
```

Sample Configuration File

```
SYBASE
```

 # Example for the direct use of code and use of the INCLUDE statement:

```
METRIC 2700
  COLLECT MW ITO "select '', 1.0 * sum(size) from master..sysusages"
```

```
  REPORT 1 "select * from master..sysusages"
  REPORT 2 INCLUDE "syb700.sql"
```

 # Example for the use of stored procedure:

```
METRIC 2701
  COLLECT MW ITO "exec sp_dbspiudm2701"
METRIC 2702
  COLLECT MW ITO "exec sp_dbspiudm2702 @threshold"
METRIC 2704
  COLLECT MW ITO "select 'UDM 2704 returned -1', 1.0 * count(*)
                  from master..sysusages
                  where size > @threshold"
```

7 Special Configurations

This chapter describes special configurations apart from the standard described in previous chapters. Among those situations covered are remote database monitoring. Also covered is how to use the DB-SPI to monitor databases in environments using Sun Clustering and MC/Service Guard. Topics appear in the following order:

Monitoring Remote Databases

Remote monitoring of MS SQL Server is not supported.

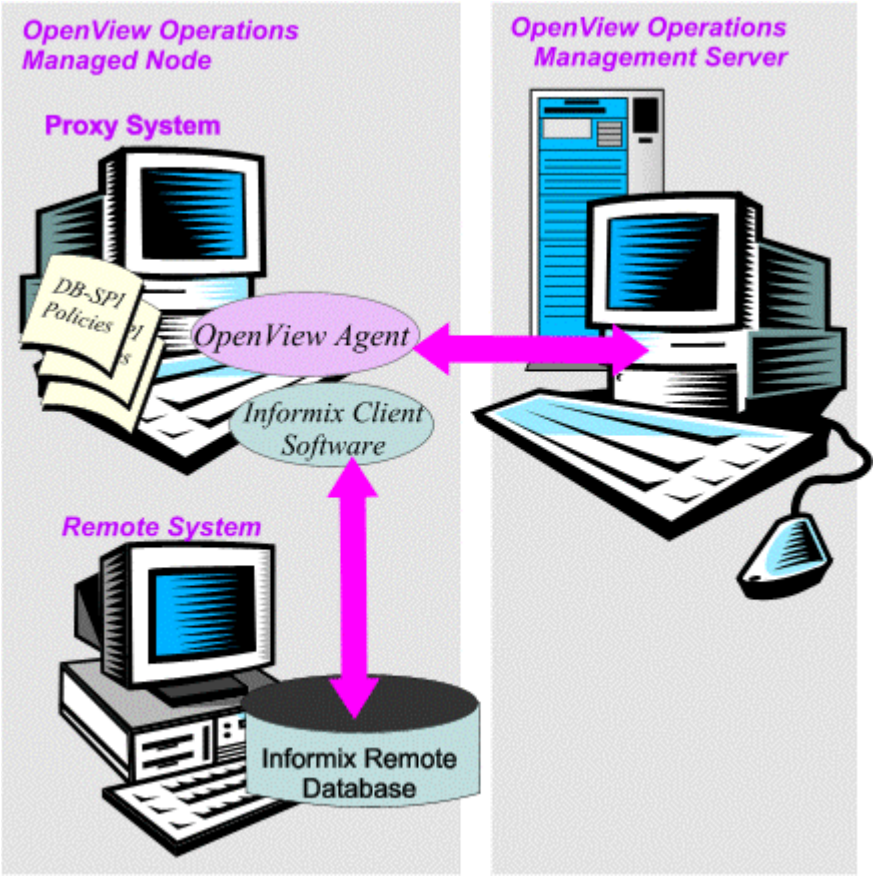
This chapter covers configurations outside the standard setups on HP-UX platforms described in earlier chapters. If you have databases running on platforms that are not supported by DB-SPI, this section shows you how you can still access the database metrics and logfiles on those systems through the database application and OVO agent software.

The *DBSPI-Oracle: Listener* template is NOT supported for remote database monitoring (databases running on unsupported platforms such as Linux).

Informix Remote Metrics

You can configure DB-SPI for Informix, with minimum effort, to remotely monitor Informix databases on systems that are not directly supported (as they are on HP-UX, Solaris, or Tru64) by DB-SPI. The steps required for remote monitoring are described below. You must use a dedicated proxy system (where no local database instance is configured) for metric data to be handled correctly.

Figure 36 Use a dedicated proxy system to access database metrics on a remote server. To do this, install DB-SPI on the same system as the database client software



Metrics 1, 2 and 70 cannot be remotely monitored. Attempting to remotely monitor these metrics will return erroneous results. To avoid this problem the DBSPI-Informix: Quick Start Rem. template group (which contains all Standard metrics except 1, 2 and 70) should be used. DBSPI-Informix: Drill Down metrics support remote databases as well.

To access the database metrics of a remote system, you can go through the connections between database client/server. With DB-SPI installed on the client, you can use the client/server connection to receive some Informix metrics. This is useful for monitoring Informix databases installed on operating systems other than HP-UX or Solaris.

Task 1: Use Informix to configure Remote/Local database Connections

In this section you configure the database remote/local database instance connections.

1. On the remote system, log on as informix and run `onstat` to make sure that the database is not running (if it is, stop it using `onmode -k`).
2. Change to user `root`.
3. Edit the `/etc/services` file and add an entry similar to the following if one does not already exist *(The service name is `online914` and should be unique in the `/etc/services` file. You later enter the same service name in the `sqlhosts` file. The socket port entry and protocol is: `1528/tcp`. Make sure that the number you choose is unique and is not used by another service. The remainder of the line are comments.)*. Your entries should like this:

   ```
   online914  1528/tcp  #Informix Online TCP/IP - remote_system
   ```

If your system uses NIS you may need to enter this information in the /etc/services file on the NIS server instead.

4. Edit the `$INFORMIXDIR/etc/sqlhosts` file so that the database uses sockets as its method of interprocess communications:

   ```
   i914  onsoctcp  remote_system  online914
   ```

(*In this example, `i914` should be the name of the database server on the machine being remotely monitored. `onsoctcp` specifies to use sockets and `tcp` as the method of interprocess communication/protocol. `remote_system` is the name of the machine on which the Informix database is running. The last column is the service name and should be the same as the entry listed in the `/etc/services` file.)*

Task 2: USE DB-SPI to Configure the Local System Connection To the Remote Database Instances

The Informix client database software and DB-SPI must be installed and configured on the same system. The database does not need to be running.

1. On the local system (where DB-SPI is installed), repeat steps **3** and **4** from **Task 1**.
2. Verify that the remote database connection is working correctly from this system by connecting to the remote database using **Check Connections.**

For further information, please refer to the *Client/Server Communications - Informix Administrators Guide.*

Task 3: Use DB-SPI to configure the remote Database Instances on the Local System

In this section you configure DB-SPI to recognize the remote database instance.

1. Double-click the **Configure DB Connections** application in the OVO Informix Application Bank and configure as usual.
2. Note the `i914` entry in the `INFORMIX` section in the figure below. Notice that the Informix home directory path is the same for both database instances in the configuration file. *Do not enter a different Home directory path for the remote server entry.*
3. (optional) If you use HP OpenView PerfView and have OpenView Performance Agent (also known as MeasureWare Agent) installed on the OVO-managed node, you can enable graphs, using the Enable Graphing application.

Task 4: Configure the Logfile Template

To access remote database logfiles, you must have OVO agent software installed on the remote system. If you do, then the next step on the local system is to configure the location of the logfile on the remote system.

1. Open the OVO Node Bank window and from the Window menu select **Message Source Templates**.
2. Create a new template group by clicking the **Add Group...** option button. If the **Add Logfile...** option button appears instead, click it and then click **Add Group...**.
3. Name the new group and description `DBSPI-Logfile <node_name>`, etc. For example:

   ```
   DBSPI-Logfile Kimball
   ```

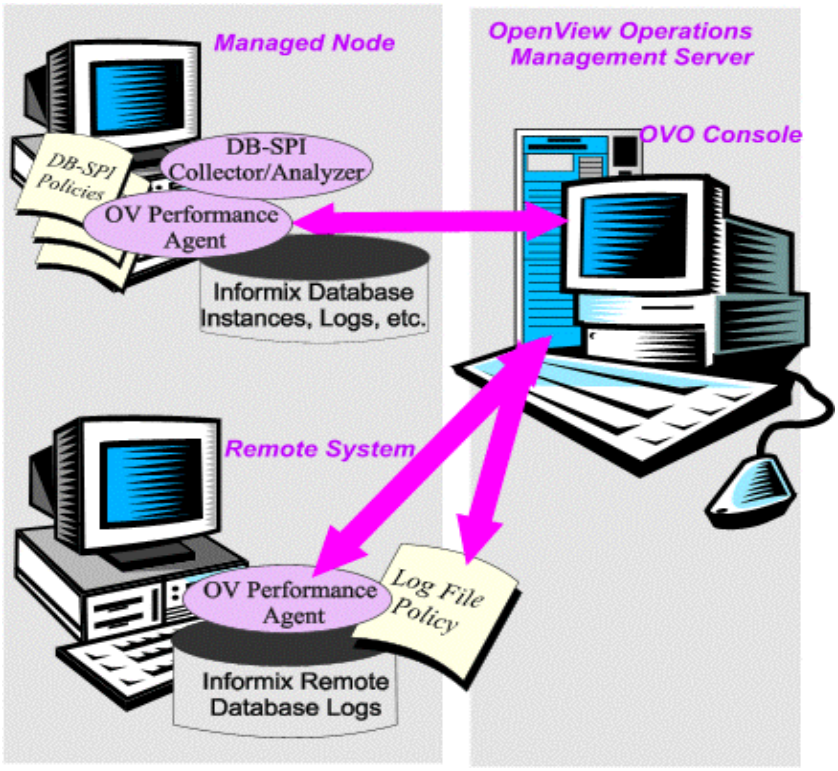
4. In the Message Source Templates window, open the **DBSPI-Informix>DBSPI-Informix Logfiles** template.
5. Select **Informix Logfiles Template** and click the **Copy...** button to copy the logfile.
6. In the Copy Logfile window in the Template Name field, enter a name for the logfile. You may want to use `<node_name>-<server> Logfile`.
   For example:

   ```
   Kimball-sales Logfile
   ```

Task 5: Deploy Remote and Logfile Templates

To monitor the Informix database on the remote system, you need to deploy the DBSPI-Informix: Quick Start Remote template group and the logfile template to the local system.

Figure 37 Informix remote system database logfile information can be accessed if an OVO agent runs on the remote system



Features Supported in Informix Remote Monitoring

Drill-down metrics: All are supported remotely. These metrics are available when you deploy the template group **DBSPI- Informix: Drill-Down**.

Reports: DB-SPI reports initiated by automatic or operator actions do not work using remote monitoring.

Applications: The following list of DB-SPI **applications** show that DB-Access works successfully, while others do not.

Table 37 Informix Applications for Remote Monitoring

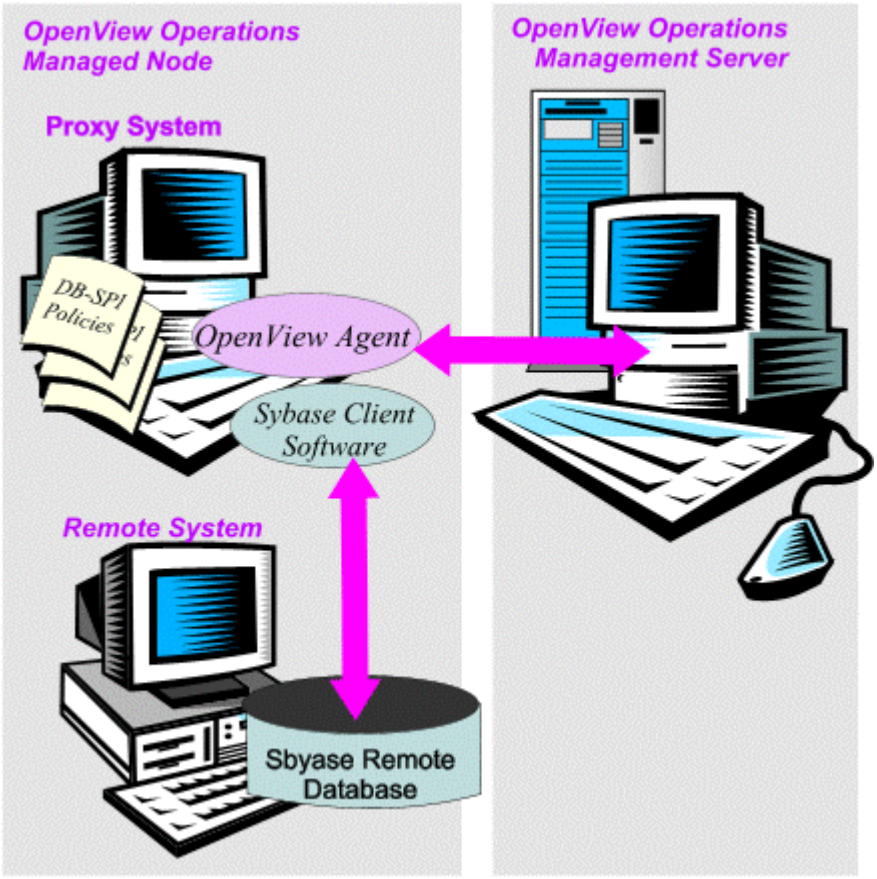
Application	Restrictions
DB-Access	Works correctly for both local and remote databases.
On-Monitor	Does not work for remote databases.
On-Stat	Does not work for remote databases.
On-Check	Does not work for remote databases.
Query Processor	Does not work for remote databases.
Start DB Server	Does not work for remote databases.
Start all Informix DB Server	Does not work for remote databases.
Stop DB Server	Does not work for remote databases.
Stop all Informix DB Server	Does not work for remote databases.
On-Perf	Does not work for remote databases.
Server Status	Does not work for remote databases.
Server Status All	Does not work for remote databases.

Sybase Remote Metrics

You can configure DB-SPI for Sybase, with minimum effort, to remotely monitor Sybase databases on systems that are not directly supported by DB-SPI (as they are on HP-UX or Solaris). The steps required for remote monitoring are described below. You must use a dedicated proxy system (where no local database instance is configured) for metric data to be handled correctly.

The steps required for remote monitoring are described in the following section.

Figure 38 Sybase Remote Monitoring System Configuration



Task 1: Use Sybase to configure Remote/Local database Connections

Remote monitoring requires that DB-SPI and a Sybase database be installed and configured on the same system. The database need not be running.

1. Open a UNIX session on the remote system and another UNIX session on the local system (the OVO-managed node).
2. In the remote system session, view the interfaces file in the $SYBASE directory and note the entry made for the database that you want to monitor.

   The interfaces file should look similar to the following.

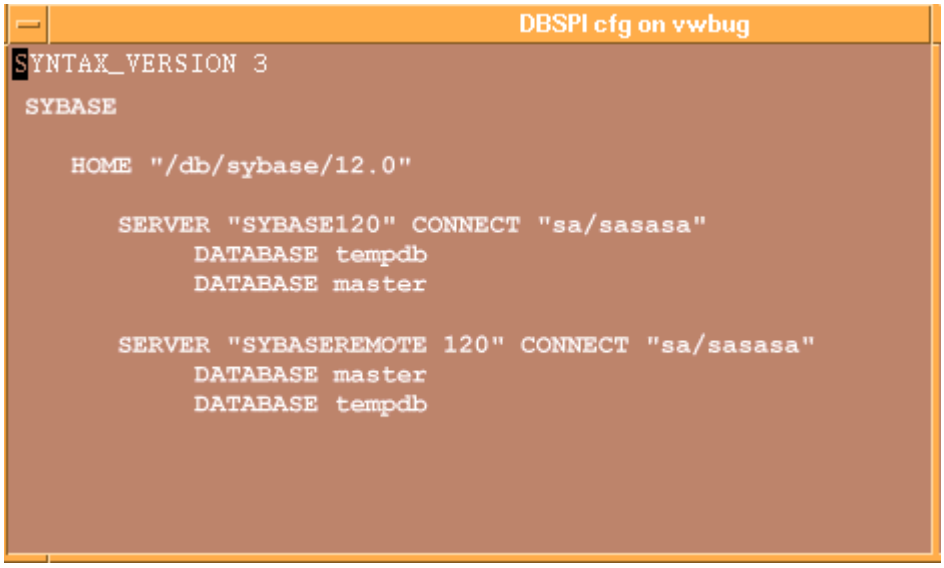
   ```
   SYBASEREMOTE115
     master tcp ether remote_system 4100
     query tcp ether remote_system 4100
   ```

3. Copy the entry corresponding to the remote Sybase database that you want to monitor into the interfaces file on the OVO-managed node.

Task 2: USE DB-SPI to Configure the remote Database Instance on the Local system

1. Use the **Configure DB Connections** application in the OVO Sybase Application Bank and configure as usual.
2. Note the `SYBASEREMOTE115` entry in the `SYBASE` section in the figure below. *Do not enter a different HOME directory path for the new server entry.*

Figure 39 Configuring Sybase DB-SPI for Remote Monitoring



Task 3: Configure the Logfile Template

To access remote database logfiles, you must have OVO agent software installed on the remote system. If you do, the next step on the local system is to configure the location of the logfile on the remote system.

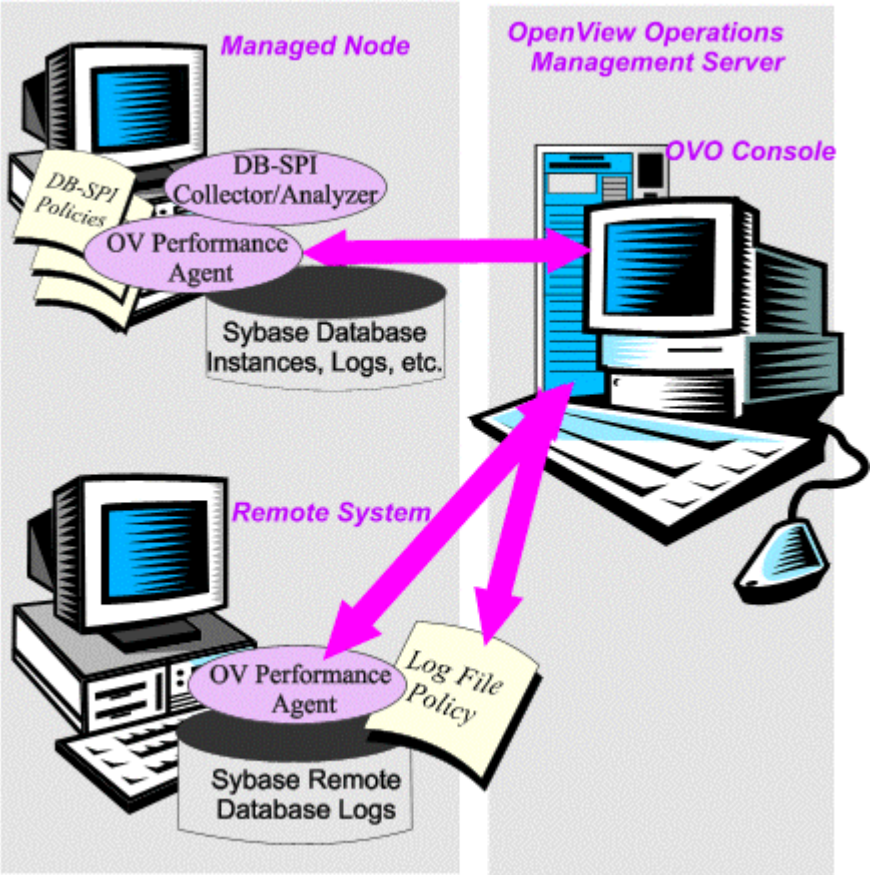
1. Open the OVO Node Bank window and from the Window menu select **Message Source Templates**.
2. Create a new template group by clicking the **Add Group...** option button.
   (If the **Add Logfile...** option button appears instead, click it and then click **Add Group...**.)
3. Name the new group and description `DBSPI Logfile-<node_name>`, etc. For example:

   ```
   DBSPI-Logfile Kimball
   ```

4. In the Message Source Templates window, open the **DBSPI-Sybase→DBSPI-Sybase: Logfiles** group.
5. Select the **Sybase Logfile Template.**
6. Click the **Copy...** button to copy the logfile.
7. In the Copy Logfile window in the Template Name field, enter a name for the logfile. You may want to use `<node_name>-<server> Logfile`.
   For example:

   ```
   Kimball-sales Logfile
   ```

Figure 40 Remote system database logfile information can be accessed if an OVO agent runs on the remote system.



Task 4: Deploy Remote and Logfile Templates

To monitor the Sybase database on the remote system, you need to deploy the DBSPI- Sybase: Quick Start Remote template group and the logfile template on the local system.

Features Supported in Sybase Remote Monitoring

Metrics: All metrics are supported remotely.

Reports: DB-SPI reports initiated by automatic or operator actions do not work using remote monitoring.

Applications: The following list of DB-SPI **applications** show those that work successfully and those that do not.

Table 38 Sybase Applications for Remote Monitoring

Application	Restrictions
isql	Works correctly for both local and remote databases.
Start DB Server	Cannot remotely start databases.
Stop DB Server	Works correctly for both local and remote databases.

Table 38 Sybase Applications for Remote Monitoring

Application	Restrictions
Server Status	The node specified is for the local machine not the remote machine. The SYBASE PATH is for the local sybase database.
Sybase Tables	Works correctly for both local and remote databases.
Free Space	Works correctly for both local and remote databases. Will not work for Sybase 11.0 (remote or local).

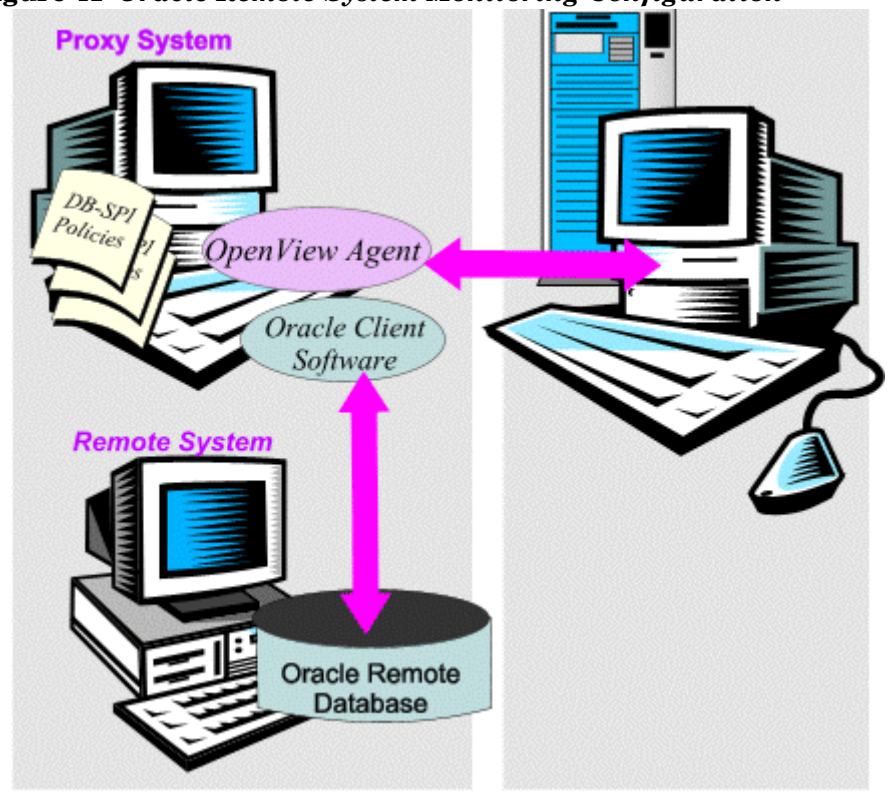
Oracle Remote Metrics

You can monitor Oracle databases remotely on systems other than those running on HP-UX, Linux [Oracle 9 only], Solaris, Tru64, AIX, or NT platforms. Each platform requires a different configuration. The first description covers systems where Oracle SQL*Net is installed.

Monitoring Oracle Databases on Systems with SQL*Net/Net8

To use DB-SPI to monitor an Oracle database remotely, you must configure DB-SPI for Oracle SQL*Net, which is an Oracle client application. This client application enables connection to the remote Oracle database. Though the remote database is running on a platform not supported by DB-SPI, you can use Oracle SQL*Net to make connections between it and the managed node. To collect the data, use a dedicated proxy system (with no local database instance configured).

Figure 41 ***Oracle Remote System Monitoring Configuration***



This section describes how to configure DB-SPI for use with Oracle SQL*Net, and provides information on the DB-SPI metrics and applications that you can used in conjunction with Oracle SQL*Net.

Throughout this section, references to SQL*Net are also relevant to Net8.

Use Oracle to configure Local/Remote system connections and Database access

Use Oracle to configure the connection between the OVO-managed node (on which DB-SPI is installed) and the remote system on which the database resides. Please see the Oracle documentation for how to use Oracle SQL*Net or Net8 to make the connection.

Task 5: USE DB-SPI to Configure the remote Database Instance on the Local system

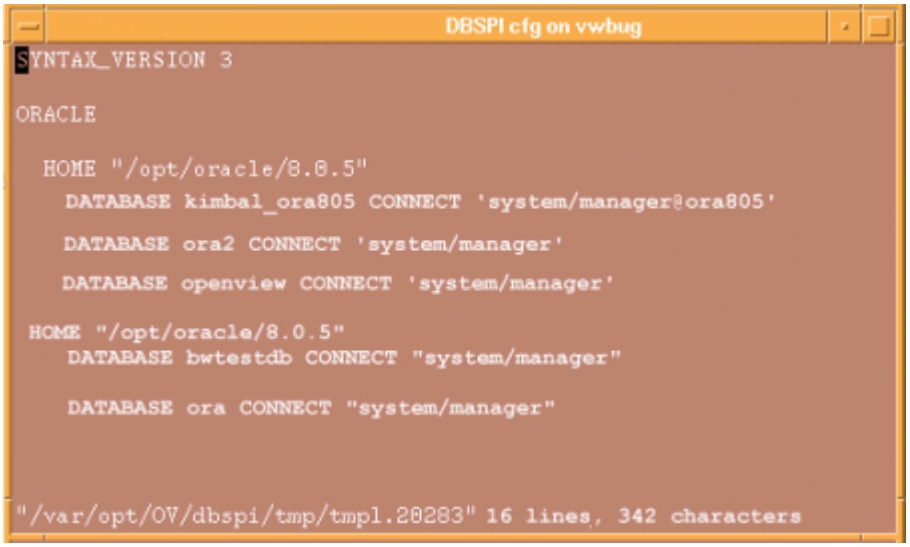
Use the Configure DB Connections application in the OVO Oracle Application Bank and configure as usual for each managed node that communicates with the remote database using Oracle SQL*Net or Net8.

1 In the configuration file include an entry for each database instance name and the user name/password metrics need to use to log onto the database.

You have probably used **Configure DB Connections** to edit the configuration file for local databases. The illustration below provides examples for local and remote databases, both of which share a similar format. However, use the following syntax when configuring remote databases:

```
username/password@oracle-alias
```

Figure 42 Sample Configuration File showing local and remote databases



In the example, `kimball_ora805` is an alias for a remote database. `@ora805` indicates that the database is remote and is accessed over Oracle SQL*Net. All other listed databases (ora2, openview, bwtestdb and ora) are local. For local databases the instance name appears after the word "`DATABASE`."

For remote databases the name that appears after the word "`DATABASE`" is the DB-SPI alias for the remote database. The name that follows "@" is the Oracle alias for the database and is specified in the `tnsnames.ora` file (you can set both the DB-SPI and the Oracle alias to the

same name). In the example illustrated, the alias `kimball_ora805` contains a prefix referencing the system name. The prefix can be used both to show the database location and also to differentiate it from any other database of the same name running on the local system.

Task 6: (optional): Integration of OpenView Performance Agent

For OVO Performance Agent integration, note the following:

1. DB-SPI alias names cannot exceed a maximum of 16 characters, and cannot be the same as the name of any local database instance.
2. The actual OVO Performance Agent collection occurs on the managed node, not the remote system. Any enabling of graphs or reports should take place on the DB-SPI managed node.

DB-SPI Metrics Supported with Oracle SQL*Net/Net8

Most DB-SPI metrics are supported for use with Oracle SQL*Net (a list of exceptions is provided below). The following message source template group containing the supported DB-SPI metrics is provided to facilitate the distribution and use of DB-SPI metrics with Oracle SQL*Net:

- **DBSPI-Oracle Quick Start Remote:** This group contains templates for monitoring Oracle databases running on remote systems using SQL*Net.
- **DBSPI-Oracle: Oracle 8 & Higher.** Contains templates that monitor metrics included only in Oracle version 8 or higher.

To use DB-SPI with Oracle SQL*Net, assign and distribute the above group to the desired managed node(s). Instructions for setting up the configuration file for the managed node(s) is provided later in this document.

DB-SPI Metrics Not Supported with Oracle SQL*Net

The following table lists the DB-SPI metrics that are not supported (or are only conditionally supported) for use with Oracle SQL*Net:

Table 39 Metrics not Supported with SQL*Net

Metric	Special Condition
DBSPI-0002	Unsupported with Oracle SQL*Net
DBSPI-0003	Works only if autoextend is turned off
DBSPI-0006	Works only if autoextend is turned off
DBSPI-0016	Works only if autoextend is turned off
DBSPI-0056	Unsupported with Oracle SQL*Net
DBSPI-0058	Unsupported with Oracle SQL*Net
DBSPI-0062	Unsupported with Oracle SQL*Net
DBSPI-0063	Unsupported with Oracle SQL*Net
DBSPI-0064	Unsupported with Oracle SQL*Net

Table 39 Metrics not Supported with SQL*Net

Metric	Special Condition
DBSPI-0065	Unsupported with Oracle SQL*Net
DBSPI-0066	Unsupported with Oracle SQL*Net
DBSPI-0203	Works only if autoextend is turned off
DBSPI-0206	Works only if autoextend is turned off
DBSPI-0216	Works only if autoextend is turned off

DB-SPI Oracle Applications Supported with Oracle SQL*Net

The applications shown below are located in the Application Bank→DBSPI→Oracle group.

Figure 43 Oracle Applications allow you to perform actions from the OVO console that are accessible on the managed node, such as starting and stopping specific database instances

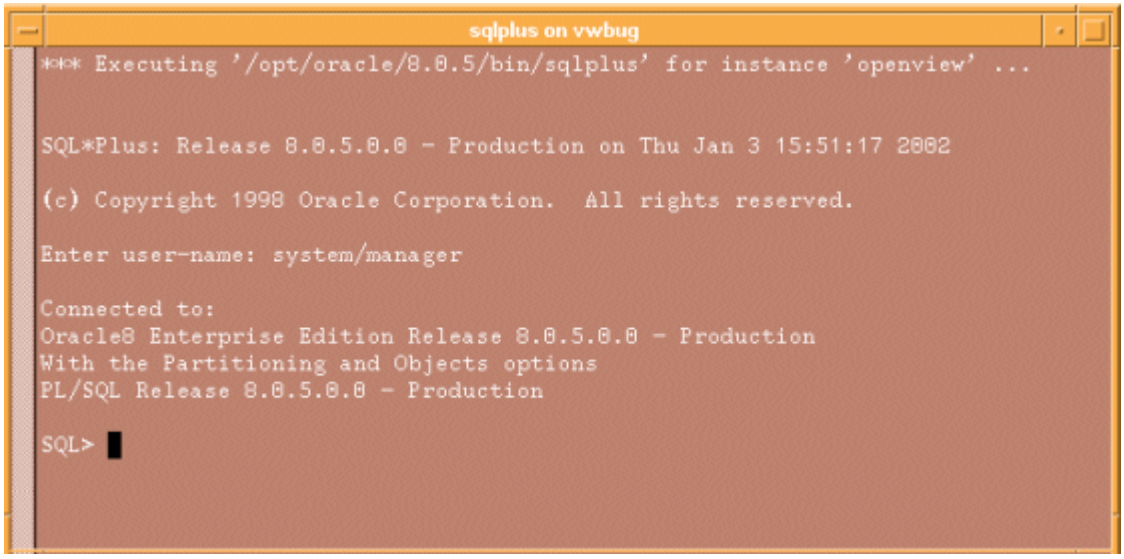
The following Oracle applications work over Oracle SQL*Net/Net8 and can be run from OVO.

- SQLPlus
- SvrMgrl - Server Manager
- SvrMgrm - Server Manager Graphical
- Import
- Export

Specifying the Oracle Alias at Log on

Even if a user has already selected a DBSPI alias for the remote database instance, the Oracle alias must still be specified at log on. For example, at the SQLPlus logon prompt the user would enter something similar to the following.

Figure 44 Specifying the Oracle Alias



Oracle Applications Not Supported with Oracle SQL*Net

The following programs in the OVO Application Bank do not work over Oracle SQL*Net:

- Start Instance
- Start all Inst.
- Shutdown Inst.
- Shutdown all Inst.
- Shut. Inst Immediate
- Shut Inst. Abort
- SQL Net Status
- SQL Net Start
- SQL Net Stop
- LSNRCTL

Configuring Remote Oracle Logfiles

Even though DB-SPI runs only on HP-UX, Linux [Oracle 9 only], Solaris, AIX, Tru64, and NT systems, you can still monitor an Oracle logfile on a remote OVO-managed system running on another platform. In the procedure that follows, at the OVO-managed console you configure, assign, and deploy the logfile template to the remote system.

Configure the logfile template/Deploy Templates

To access remote database logfiles, you must have OVO agent software installed on the remote system. If you do, the next step on the local system is to configure the location of the logfile on the remote system.

1 Open the OVO Node Bank window and from the window menu select **Message Source Templates**.

2 Create a new template group by clicking the **Add Group...** option button. (If the **Add Logfile...** option button appears instead, click it and then click **Add Group...**.)

3 Name the new group and description `DBSPI Logfile-<node_name>`, etc. For example:

```
DBSPI-Logfile Kimball
```

4 In the Message Source Templates window, open the **DBSPI-Oracle→DBSPI-Oracle Logfiles** template.

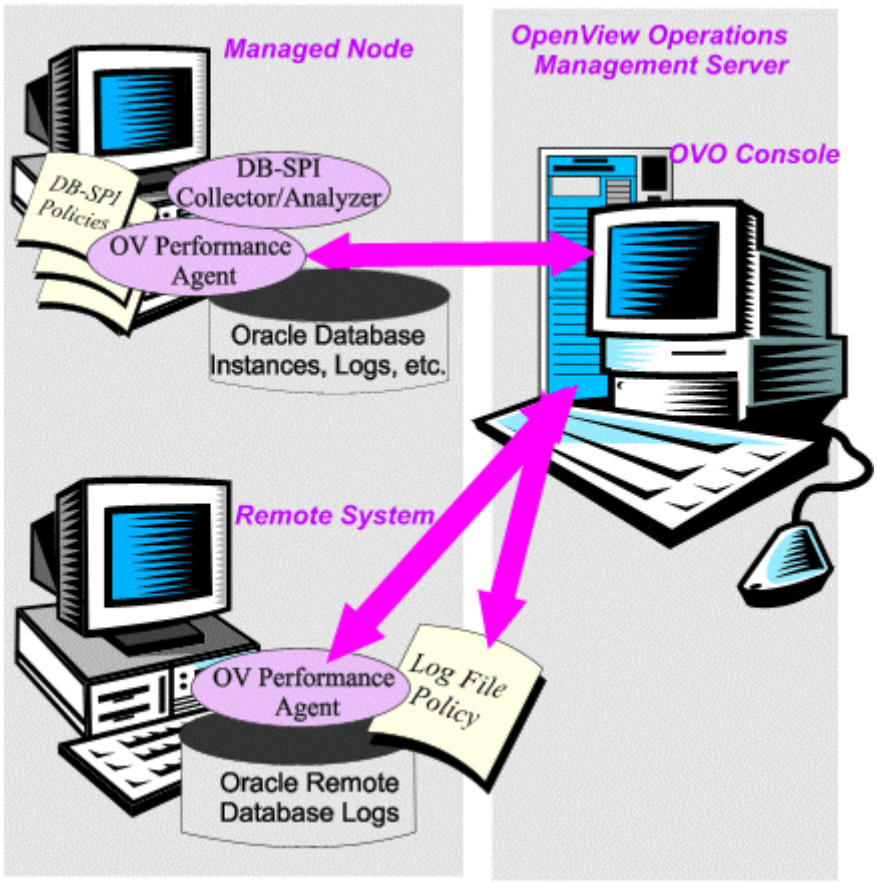
5 Click the **Copy...** button to copy the logfile.

6 In the Copy Logfile window, in the Template Name field, enter a name for the logfile. You may want to use `<node_name>-<server> Logfile`. For example:

```
Kimball-sales Logfile
```

7 Deploy the **DBSPI- Oracle: Quick Start Remote** template group and the logfile template on the local system.

Monitoring Oracle Remote Logfiles, System Configurations



Oracle Data Guard

For Oracle 9.x and later installations, the Database SPI automatically detects the Oracle Data Guard environment. The DB-SPI metrics for Data Guard monitor services such as log transport services, log apply services, and standby destination errors. The metrics monitor these services and a message is sent to the message browser when the monitored value matches the threshold condition.

Implementing the metric collection requires no manual configuration. You need only deploy Oracle template groups as you would to any managed node, and DB-SPI does the following:

- Detects the Data Guard environment.
- Identifies the type of the database, such as primary database, physical standby database, or logical standby database.
- Targets all database instances (including the first) for remaining metric collections.
- Starts data collection.
- Sends alarms with appropriate message.

Oracle RAC (Real Application Clusters) Environments

For Oracle 9.x or higher installations, the Database SPI can automatically detect databases running in clusters. When such configurations are detected, the Database SPI identifies one active node for global metric collection. Metrics designated as global generate alarms only from this selected node.

For example, because storage is shared in a RAC cluster, storage metrics need to run only on a single node. Other metrics, producing different values on each node, will run on all nodes. As a result, no redundant, unnecessary messages (for global metrics) appear in the OVO message browser.

Oracle 8.x requires some manual configuration (see Oracle 8.x OPS Environments (setup required).

Monitoring Oracle 9.x & Higher (no setup)

Implementing the shared/individual metric collections requires no manual configuration. You need only deploy Oracle template groups as you would to any managed node, and the DB-SPI does the following:

- Detects the cluster configuration.
- Targets the first active instance (system with the lowest instance ID in the Oracle `GV$_ACTIVE_INSTANCES` table) for the global metric collection.
- Targets all database instances (including the first) for remaining metric collections.
- Starts data collections.
- Should the database instance designated for global-collection become inactive, global collection automatically moves to the next active instance in the table, and global metric collection resumes.

Manually configuring RAC

You can turn global database metrics on/off, using an Oracle application designed for that purpose. The application, located in the DBSPI-Oracle group has three possible settings:

- **On**: turns global metric collection on for the selected node
- **Off**: turns global metric collection off for the selected node
- **Auto**: allows the Database SPI to automatically turn global collection on/off, depending on active/inactive database instances listed in the Oracle system table.

If you decide to change the present metric collection configuration, you need to turn global metrics *Off* on the currently designated node; otherwise, global metrics are collected on two nodes and you will see redundant messages in the Message Browser (for the duplicate global metric collections).

To manually turn global metric collection on/off:

1. From the Window menu open the **Application Bank** and select **DB-SPI→Oracle**.
2. From the Window menu open the **Node Bank**.
3. Select a node and drag and drop it on a **RAC Global Metrics** (*UNIX for UNIX nodes, or Windows for Windows nodes*) application.
4. In the window that appears, choose the desired state from:
   On
   Off
   Auto

You can always revert to *Auto* if you decide you no longer want to manually designate global metric collections; just be sure to first use the RAC Global Metrics application to turn the current global collection to Off. Then use *Auto* for any node.

Table 40 Oracle RAC Global Metrics

Global Metrics
4, 5, 6, 7, 9, 11, 16, 17, 18, 42, 47, 67, 77, 78, 79, 80, 81, 97, 121, 122, 125, 203, 206, 210, 212, 215, 216, 217, 218

Monitoring Oracle 8.x (setup)

Monitoring DB-SPI in Oracle environments with OPS (Oracle Parallel Server) requires manual configuration. To avoid problems in connecting to the database or collecting metric data, you must modify some Oracle settings and some Database SPI files.

By setting four OPS Group Membership Service- (GMS-) related environment variables in the OVO agent startup file, you ensure that the *DB-SPI Connection* application and metric processing work. Also, you can avoid errors when running applications in the DB-SPI Application Bank and metric reports by setting the four OPS GMS environment variables in the `root /etc/profile` file.

Task 1: Edit the OVO Agent Startup File and the Root User Startup Profile

Completing the tasks that follow allows DB-SPI to connect to an OPS database. Without these changes, the following Oracle error occurs whenever DB-SPI tries to connect to the OPS database:

```
ORA-29701: unable to connect to Group Membership Service (GMS)
```

Edit the OVO Agent Startup file

1 Open the OVO agent startup file (the location of this file depends on the type of UNIX system:
HP-UX: `/etc/rc.config.d/opcagt`
Linux: `/etc/rc.config.d/opcagt`
Solaris: `/etc/init.d/opcagt`
AIX: `/etc/rc.opc`

2 Add the following four Oracle Parallel Server Group Membership environment variables:

```
export GMS_HC_SOCKET=<file_path>
export GMS_LOG_DIR=<file_path>
export GMS_NODE_LIST=<file_path>
export GMS_QUERY_FILE=<file_path>
```

Example:

```
export GMS_HC_SOCKET=/tmp/serv.hc
export GMS_LOG_DIR=/tmp/.ogms
export GMS_NODE_LIST=/u01/app/oracle/product/805/gms/nodes.lst
export GMS_QUERY_FILE=/dev/rogms
```

Edit the Root User Startup File

1 Open the system-wide profile file in `/etc/profile`.

2 Add four Oracle Parallel Server Group Membership environment variables:

```
export GMS_HC_SOCKET=<file_path>
export GMS_LOG_DIR=<file_path>
export GMS_NODE_LIST=<file_path>
export GMS_QUERY_FILE=<file_path>
```

On some systems the `/etc/profile` file is automatically generated from other files, so you may need to edit another file to make this change permanent. See your system administrator for details.

3 Restart the OVO agent by entering:

```
/opt/OV/bin/OpC/opcagt -kill
/opt/OV/bin/OpC/opcagt -start
```

Task 2: Designate metric collections for each system in the cluster.

For databases running in cluster environments, the Database SPI includes an application for Oracle that allows you to designate how you would like to implement collection of shared (global) metrics.

For an explanation of global versus non-global metrics, see the previous section, Oracle RAC (Real Application Clusters) Environments on page 165.

Using this application for Oracle 9, you can choose *Auto* mode and the Database SPI automatically implements both global and non-global collections. But because the *Auto* feature is not supported for Oracle 8.x, you must manually turn global metric collection *On* for one node and then run the same application in *Off* mode on all remaining nodes in the cluster.

To implement cluster metric collections in an OPS environment:

1. From the Window menu open the **Application Bank** and select **DB-SPI→Oracle**.
2. From the Window menu open the **Node Bank**.
3. Select one node and drag and drop it on the **RAC Global Metrics** application.
4. In the window that appears, choose **On.**
5. Repeat step #3 for all remaining nodes in the cluster, but choose **Off**.

Monitoring Oracle Enterprise Manager

The Databases SPI can be configured to capture the alerts and events generated by Oracle Enterprise Manager 10g Grid Control and send the information to OVO. This information is displayed on the OVO console, with the specified severity level.

Before you can monitor Oracle Enterprise Manager 10g Grid Control installations using the Database SPI, you must configure the Oracle Enterprise Manager 10g Grid Control application to send alerts and events to OVO.

To configure Oracle Enterprise Manager 10g Grid Control, complete the following steps:

1. Log on to Oracle Enterprise Manager 10g Grid Control console.
2. Click **Setup**, followed by **Notification Methods**.
3. In the `Scripts and SNMP Traps` section, select `OS Command` to add a method from the pull down list and click **Go**. You must provide the name of the OS command used to forward the notifications to OVO console.

   Example:

   Name: HP OpenView Integration

   Description: OS Command Integration to send Alerts/Events to HP OpenView console

   OS Command: `/var/opt/OV/bin/instrumentation/dbspicaoem.sh`

For a Windows managed node, provide the entire Perl path, followed by the entire path of the script `dbspicaoem.pl` deployed from the server, in the DOS format and using ~ if spaces appear.

Example:

```
C:/oracle/product/10.1.0/agent/perl/5.6.1/bin/MSWin32-x86/
perl.exe C:/Program Files/HP OpenView/Data/bin/instrumentation/
dbspicaoem.pl
```

4. Click **Preferences** and select `Rules`.
5. Edit the rules from which you want to receive the event.

6 Assign method to OS Command for each of the edited rules.

The test message that is generated when you click **Test OS Command** on the Oracle Enterprise Manager 10g Grid Control browser, is not reported on the OVO browser.

Setup for Clustered, Fail-over Environments (all database types)

This setup relates only to fail-over cluster environments. Do not use with Oracle RAC or Oracle OPS.

The Database SPI can be configured to accommodate cluster environments where fail-overs allow uninterrupted database availability. When you configure the Database SPI to be synchronized with a cluster environment, monitoring can be made to switch off for a failed node and switch on for an active node. For recognizing clustered database instances, DB-SPI relies on two `XML` configuration files. These files allow the OVO agent to automatically enable instance monitoring on the currently active node after disabling instance monitoring on the inactive node.

The DB-SPI setup for a cluster environment requires that you do the following:

- (if needed) Modify the `XML` file included with the Database SPI (see page 170).
- (if not done already) Configure each database instance on every node where the instance runs; then deploy DB-SPI policies on the configured nodes (see page 173).
- Create an `XML` file that associates DB-SPI-monitored instances with the cluster packages (see page 173).

To monitor a cluster-aware application, DB-SPI requires that a monitored resource in the cluster group contain both a n*etwork name* and an *IP address*. If the cluster group does not meet these requirements, the following errors are logged in the opcerror file on the cluster node:

- "Could not perform cluster API function, error code 1008 returned System Error Number: -1 (ffffffff) - (OpC30-3223)"
- "Could not read cluster information System Error Number: -1 (ffffffff) - (OpC30-3221)"
- "Application Package Monitor of subagent 0 aborted; process got signal 1 (OpC30-1041)."

Software Requirements

The Database SPI is compatible with cluster software as follows:

- Microsoft Clustering Solutions (MSCS) on Windows
- HP Serviceguard on HP-UX nodes
- VERITAS on HP-UX and Solaris nodes
- Sun Cluster on Solaris nodes

The XML file Included with DB-SPI

The Database Smart Plug-in includes an XML file for each database type. This file works in conjunction with another file that you create in Task #3. The purpose of the file is to list all the Database SPI templates on the managed node so that these templates can be disabled/enabled as appropriate for inactive/active managed nodes.

The configuration below describes a scenario where DB-SPI is enabled for monitoring a database instance for one node in the cluster, while it remains disabled for all other nodes in the cluster.

Task 1: Modify/deploy/configure XML file as needed.

For DCE agent installations: If necessary, modify the DB-SPI XML file listing DB-SPI templates to match the renamed templates and redeploy.

The Database SPI (dbspi_<*db_type*>.apm.xml) file for your database type is normally ready to use with no configuring. However, if you have renamed any templates, you will need to modify the file.

The XML file lists all the monitor, logfile, and schedule templates for purposes of disabling/enabling. These templates are disabled only when the last instance is moved to another node. The **start/stop** commands are performed for each instance that is started/stopped on the node.

a. On the management server uncompress/open the Database SPI XML file matching your database type (see the section that follows for location):

Informix:	dbspi_informix.apm.xml
MS SQL:	dbspi_mssqlserver.apm.xml
MS SQL:	dbspi_mssqlserver_jp.apm.xml (Japanese)
Oracle:	dbspi_oracle.apm.xml
Oracle:	dbspi_oracle_jp.apm.xml (Japanese)
Sybase:	dbspi_sybase.apm.xml

b. Modify template names as necessary in the DB-SPI XML file to match those templates currently deployed on managed nodes.

c. Save the file in the monitor directory, then redeploy **Monitors**.

For HTTPS agent installations:

Note: This step unnecessary if you assigned policies to virtual nodes.
If you modified template names, follow the steps in the above procedure so that the template names in the XML file match those of the templates currently deployed on managed nodes.

d. From the Window menu, select the **Application Bank** and open the **DB-SPI** application group.

e. Open the **DBSPI-Admin** or **DBSPI-Admin (NT)** group, depending on the managed node operating system you are configuring (UNIX or Windows).

f. In the DBSPI-Admin window locate the **Cluster Config** application
or
In the DBSPI-Admin (NT) window locate the **Cluster Config (NT)** application.

g Open the **Node Bank** window and navigate to the DB-SPI node group you are configuring for cluster awareness.

h Drag and drop the desired node group onto the **Cluster Config** or **Cluster Config (NT)** application, as appropriate to the managed node operating system.

DB-SPI XML File Location & Default Content

On the OVO management server, the XML file is located for the various node operating systems/database types as follows:

```
ENGLISH
Windows DCE agent: /var/opt/OV/share/databases/OpC/mgd_node/customer
  /ms/intel/nt/monitor/
      dbspi_mssqlserver.apm.xml.
      dbspi_oracle.apm.xml.

Windows HTTPS agent: /var/opt/OV/share/databases/OpC/mgd_node/customer
  /ms/x86/winnt/monitor/
     dbspi_mssqlserver.apm.xml
     dbspi_oracle.apm.xml

Solaris DCE agent:
/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc
   /solaris/monitor/
      dbspi_informix.apm.xml
      dbspi_oracle.apm.xml
      dbspi_sybase.apm.xml

Solaris HTTPS agent: /var/opt/OV/share/databases/OpC/mgd_node/customer/
sun/sparc
  /solaris7/monitor/
     dbspi_informix.apm.xml
     dbspi_oracle.apm.xml
     dbspi_sybase.apm.xml

HP-UX 11.x DCE agent
/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/pa-risc
   /hp-ux11/monitor/
      dbspi_informix.apm.xml.Z
      dbspi_oracle.apm.xml.Z
      dbspi_sybase.apm.xml.Z

HP-UX 11.x HTTPS agent: /var/opt/OV/share/databases/OpC/mgd_node/customer/
hp/pa-risc
  /hpux1100/monitor/
     dbspi_informix.apm.xml
     dbspi_oracle.apm.xml
     dbspi_sybase.apm.xml

JAPANESE:
Windows DCE agent:
/var/opt/OV/share/databases/OpC/mgd_node/customer/ms/intel
   /nt/monitor/
      dbspi_mssqlserver_jp.apm.xml
      dbspi_oracle_jp.apm.xml
```

Windows HTTPS agent: /var/opt/OV/share/databases/OpC/mgd_node/customer/ms/x86
  /winnt/monitor/
    dbspi_mssqlserver_jp.apm.xml
    dbspi_oracle_jp.apm.xml

Solaris DCE Agent:
/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc
   /solaris/monitor/
      dbspi_informix.apm.xml
      dbspi_oracle_jp.apm.xml
      dbspi_sybase.apm.xml

Solaris HTTPS Agent:
/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc
  /solaris7/monitor/
     dbspi_informix.apm.xml
     dbspi_oracle_jp.apm.xml
     dbspi_sybase.apm.xml

HP-UX 11.x DCE Agent
/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/pa-risc
   /hp-ux11/monitor/
      dbspi_informix.apm.xml
      dbspi_oracle_jp.apm.xml
      dbspi_sybase.apm.xml

HP-UX 11.x **HTTPS Agent:**
/var/opt/OV/share/databases/OpC/mgd_node/customer/
     dbspi_informix.apm.xml
     dbspi_oracle_jp.apm.xml
     dbspi_sybase.apm.xml

Example (uses MS SQL Server configuration)

```
<?xml version="1.0" ?>
   <APMApplicationConfiguration xmlns="http://www.hp.com/OV/opcapm
      /app">
     <Application>
        <Name>dbspi_mssqlserver</Name>
          <Template>DBSPI-MSS-05min-Reporter</Template>
          <Template>DBSPI-MSS-1d-Reporter</Template>
          <Template>DBSPI-MSS-05min</Template>
          <Template>DBSPI-MSS-15min</Template>
          <Template>DBSPI-MSS-1h</Template>
          <Template>DBSPI-MSS6-05min</Template>
          <Template>DBSPI-MSS6-15min</Template>
          <Template>DBSPI-MSS6-1h</Template>
          <Template>DBSPI Microsoft SQL Server</Template>
          <StartCommand>dbspicol ON $instanceName
          </StartCommand>
          <StopCommand>dbspicol OFF $instanceName</StopCommand>
     </Application>
   </APMApplicationConfiguration>
```

Task 2: Configure instances to monitor.

If you have not already done so (in Chapter 2, Installing and Configuring the Database SPI), configure each database instance connection and deploy the Database SPI templates and configuration file on every node on which a fail-over may occur.

1 Use the DB-SPI Admin application **Configure DB Connections** to configure each database instance for all applicable nodes.
(Refer to Appendix C for and ignore error messages where connections cannot be made to inactive cluster nodes.)

2 **A. For managed nodes using DCE agents:**
Deploy the DB-SPI **Quick Start** template group and/or any template you want to use on each managed node.

B. For managed nodes using HTTPS agents (available in OVO 8.x):
To deploy DB-SPI policies where virtual nodes are *not used*, follow the same procedure used for DCE agents. Otherwise, to deploy DB-SPI policies where virtual nodes are used, first add the Virtual Node(s); then deploy the DB-SPI **Quick Start** template group as follows:

a In the Node Bank select **Actions→Node→Add**.

b Type the name of your virtual node.

c Select **HTTPS** as the managed node type and press the **Advanced options...** button.

d In the Node Advanced Options window, check **Cluster Virtual Node** and enter the HA resource group name and all the physical nodes in the cluster

e Click the **Close** button; then the **OK** in Add Node window.
(The newly created Virtual Node should appear in the Node Bank).

f Assign and deploy the **DB-SPI Quick Start** template group to the Virtual Node.

The XML File You Create

The second XML file is one that you create and save as: apminfo.xml This file, working in conjunction with the Database SPI-included XML file, allows you to associate DB-SPI monitored instances with cluster packages. As a result, when a package is moved from one node in a cluster to another node in the same cluster, monitoring stops on the failed node and starts on the new node.

Task 3: Create the apminfo.xml file (describes cluster instances).

The examples below show DB-SPI instances for Informix, MS SQL Server, Oracle and Sybase. Remember, you must name the file `apminfo.xml`.

1 Using a text editor, create a file with entries as specified below.

In the file, enter the *Application Name* to match the prefix of the apm.xml file (for example, for Oracle, you would enter dbspi_oracle, as shown below). Enter the *Instance Name* to match the instance name entered in the DB-SPI configuration file.

Table 41 XML File Entries

Database Type	XML File <Application> <Name>	XML File <Instance> <Name>
Informix	dbspi_informix	Insert the entry made to the configuration file (Appendix C) for: <*server_name*>.
MS SQL Server	dbspi_mssqlserver or for Japanese systems: dbspi_mssqlserver_jp	Insert the entry made to the configuration file (Appendix C) for: <*server_name*>\<*instance_name*>
Oracle	dbspi_oracle or for Japanese systems: dbspi_oracle_jp	Insert the entry made to the configuration file (Appendix C) for: <*oracle_server_instance_name*>
Sybase	dbspi_sybase	Insert the entry made to the configuration file (Appendix C) for: <*server_name*>

User-created XML file with entry descriptions

The table below shows the cluster software name that maps to the text you insert between "`<Package></Package>`."

Table 42 Specific Cluster Software Package Terminology

Cluster software	Package name component
Microsoft Cluster Server	Group
MC/Service Guard	Package
VERITAS Cluster Server (HP-UX and Solaris)	Service Group
Sun Cluster	Resource Group

```
<?xml version="1.0" ?>
     <APMClusterConfiguration>
          <Application>
                <Name>dbspi_mssqlserver</Name>
                <Instance>
                       <Name>MSSQLA</Name>
                       <Package>MSSQLA-VS1</Package>
                </Instance>
                <Instance>
                       <Name>MSSQLB</Name>
                       <Package>MSSQLB-VS2</Package>
                </Instance>
          </Application>
     </APMClusterConfiguration>
```

The Instance <*Name*> is the SQL Virtual Name and the <*Package*> is the Group Name.

2 Save the completed `apminfo.xml` file on each node in the cluster in the directory specified below:

— On HP-UX or Solaris using DCE agents: `/var/opt/OV/conf/OpC/`
On HP-UX or Solaris using HTTPS agents: `/var/opt/OV/conf/conf`

— On Windows nodes using DCE agents:
<*installation_drive*>`\usr\OV\conf\OpC\`
On Windows nodes using HTTPS agents:
<*installation_directory*>`\data\conf\conf\`

In some cases the above directories may not exist; if so, create the directories on the managed node.

For UNIX clusters, deploy the OVO cluster template(s) to all physical nodes in the cluster as follows:
For HP Serviceguard on HP-UX nodes:
Syslog (ServiceGuard)
For Sun Cluster on Solaris nodes
Engine Log (SC)
For VERITAS on HP-UX and Solaris nodes:
Engine Log (VCS)
Engine Notify Log (VCS)

The *VCS Engine log template* may need to be modified to read the log files correctly. The "Engine Log (VCS)" logfile template that is used to monitor VERITAS cluster fail-over should have conditions starting with "`^TAG<*>`". If your installation has conditions that start with only "`^TAG_C`" or "`^TAG_D`", modify the conditions to use the <*> wildcard instead of C and D.
For example, change "`^TAG_C<4#>/<2#>/<2#><S><2#>:<2#>:<2#><S>VCS:<*>:Group <@.group> is online on system <@.system>`"
to
"`TAG_<*> <4#>/<2#>/<2#><S><2#>:<2#>:<2#><S>VCS:<*>:Group <@.group> is online on system <@.system>`"

3 On each node, stop and restart the agent by running the following commands:

```
opcagt -kill
opcagt -start
```

HP Serviceguard Installations with Earlier OVO Agents

The section that follows is relevant to Database SPI installations using earlier OVO agents, prior to version 7.1. In addition, if you use Oracle RAC or Oracle Parallel Server, *do not use this method*.

This section describes how to set up and configure DB-SPI if OVO or one or more database instances are in an HP Serviceguard virtual package.

Task 1: Complete Setup Procedures

The following steps outline the differences between a standard DB-SPI installation (see Chapter 2 of this manual) and installation if one or more database instances are in a HP Serviceguard virtual package.

If OVO is a package in MC/Service Guard, complete the first step; if not, start at step 2.

1. Install DB-SPI on both nodes of the MC/Service Guard cluster following the section titled Install on the Management Server on page 22. *(Do not distribute or configure DB-SPI on any managed nodes until you have installed DB-SPI on both nodes of the OVO MC/Service Guard cluster.)*
2. Open the Message Source Templates window and double-click the **SPI for Databases** group.
3. Double-click the **DB-SPI**<*application_name*> template group that you are using to show the templates available in the group.
4. Click **DB-SPI <application_name>: Logfiles** and select the **Modify** button to open the Modify Logfiles window.
5. In the Monitoring Options section of the window deselect the **Message on No Logfile** option to prevent error messages from appearing in the OVO Message Browser.
6. Assign/distribute the DB-SPI templates and files (actions, commands, monitors) to all HP Serviceguard cluster nodes capable of running the Oracle package as described in Deploy Actions/Monitors/Commands to DB-SPI Node Group(s) on page 24.
7. Configure DB-SPI as described in Installing and Configuring the Database SPI on page 21.
8. Distribute the configuration to all HP Serviceguard nodes.
9. The DB-SPI Collector/Analyzer suspends collection without reporting errors if the file `defaults` file contains the entry:

```
COLLECTION OFF

DBSPI provides a utility script to trigger entries in this file and
report a change in state to the OVO Message Browser. If the file is
non-existent or contains other keywords, collection is enabled.
```

An OVO 4.0 agent requires the `/var/opt/OV` directory on a file system local to the cluster.

To turn some collections off, while leaving others on: The `defaults` file must be located on the managed node where you want to make the change. If this file is not on the managed node, create it. For the location of the `defaults` file, see Data, Log, and Configuration Files **on page 238**

On the managed node, edit the defaults file to contain the line:
<*first_database_instance_name*> OFF
<*second_database_instance_name*> OFF

Entry Examples
ora803 OFF
openview ON
SAP1 OFF

COLLECTION OFF overrides any other entries in the file.

Task 2: Modify the Oracle Package Control Script

The package control script is typically located in the following:

/etc/cmcluster/<PKG-name>/control.sh.

The script location is defined in the Oracle Package ASCII file (refer to the HP Serviceguard documentation for details).

Use uppercase ON and OFF function.

1 Add the following commands to the RUN and HALT script of the Oracle control.sh file at the end of these functions.

```
function customer_defined_run_cmds

{...

(Run the command to start the database.)

/var/opt/OV/bin/OpC/cmds/dbspicol ON

...}

function customer_defined_halt_cmds

{...

/var/opt/OV/bin/OpC/cmds/dbspicol OFF

(Run the command to shut down the database.)

...}
```

2 Set the statements at the beginning of the halt/run functions to:

prevent overriding return values from other actions

It is not necessary to run cmapplyconf after changing the control.sh file

3 Perform a test transfer of the Oracle package to all standby cluster nodes using a Maintenance window to verify the correct functioning of the DB-SPI Collector/Analyzer.

Currently no other settings are used for the `defaults` file, so it is safe to remove it from the run sequence. However, future releases of DB-SPI might contain additional keywords requiring the halt and run scripts to toggle the `COLLECTION` directive rather than removing the `defaults` file.

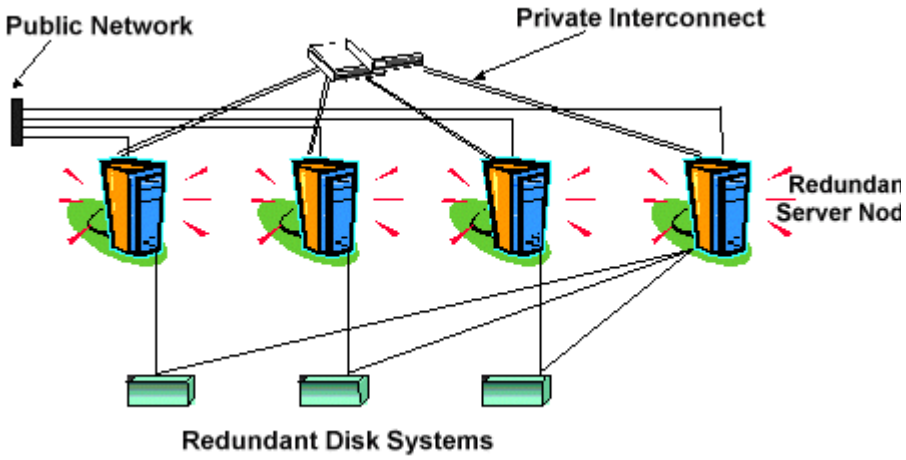
Sun or VERITAS Clustered, Fail-over Environments with 7.1 or Earlier Agents

The section that follows is relevant to Database SPI installations using earlier OVO agents, prior to version 7.1.

The Database SPI can be used in Sun or VERITAS Clusters with minimal configuring and changes to the standard installation tasks. This section includes those tasks and the syntax to use for Sun or VERITAS programs. If you run Oracle in a clustered environment configured with either Sun Clusters or VERITAS, please follow the installation/configuration steps included in this chapter, rather than the standard installation contained in Chapter 2.

OpenView Operations in a Cluster Environment

The OpenView Operations for UNIX documentation includes detailed information on how OVO recognizes logical hosts and their disk groups. Please refer to these documents for background information and to ensure that your configuration for OpenView Operations has taken into account clustering issues such as logical and physician/master hosts and shared disk groups.



Compatibility Requirements for Clustering

To operate the Oracle DB-SPI in a configuration that uses either Sun Cluster or VERITAS, your systems must meet the following requirements:

— Solaris operating system on two or more SPARC platforms

— Compatible Sun Storage

— OVO agent software installed on all physical Sun Cluster nodes.

— Sun Cluster software including the required patches
or
VERITAS Volume Manager Storage Administrator

Set Up the Database SPI to Handle Cluster Software

The OVO management server must be running on the cluster node when you run **swinstall**. During the installation you receive warnings about existing template groups, templates, conditions, which you can ignore. Warnings are listed in the `swagent.log` file. Refer to the manual for the details of the steps listed below.

If OVO is a package in the Sun Cluster, complete the Tasks 1 through 3; if not, start at Task 4.

Task 1: Install the DB-SPI on the management server

1 Log on as root, and open a terminal window.

2 In the terminal window, verify that the DISPLAY environment variable is set to your terminal window by entering: `echo $DISPLAY`
(If the DISPLAY environment variable is not set correctly, set it to your terminal window).

3 Run **swinstall** from the command line; for example, to install all bundles, you would enter (on a single line):
`/usr/sbin/swinstall -s /dvdrom/DB_SPI_10.40.000.sdtape DBSPI`

Substitute the software bundle name for the DB-SPI, see Table 2 on page 23.

4 In the Software Selection window, open DB-SPI to optionally proceed to the subproduct or fileset level or to install all filesets.

5 Highlight the desired filesets and mark them for installation by selecting **Actions→Mark for Install** from the menu.

6 Select **Actions→Install(analysis)** to complete the installation.

Task 2: Verify successful installation

1 Click the **Logfile** button to verify that the analysis phase of the installation completes without errors. If the analysis phase succeeds, begin the installation as follows

a In the Logfile window click **OK**.

b In the Install Analysis window click **OK**.

c In the Confirmation window select **Yes**.

If the analysis phase fails, follow the recommendations made in the logfile (located in the /var/adm/sw/swagent.log) and rerun the analysis.

The software installation execution phase begins.

2 To confirm that the installation completes without errors, click the **Logfile** button.
If the installation execution phase succeeds:

a In the Logfile window click **OK**.

b In the Install window click **Done**.

If the installation phase fails, follow the recommendations made in the logfile (located in `/var/adm/sw/swagent.log`) and execute the installation again.

If you run the OVO management server as a package within the Sun cluster environment, repeat Tasks 1 and 2 on all cluster nodes that are capable of running the package. Do not distribute or configure the DB-SPI on any managed nodes until you have installed the DB-SPI on all nodes in the OVO Sun cluster.

Task 3: Modify the Message Source Template

1. Open the Message Source Templates window and double-click **SPI for Databases**.
2. Double-click the **DB-SPI**<*database_type*> template group you are using to expand the group and view available templates.
3. Click **DB-SPI**<*database_type*> and **Logfiles,** and select the **Modify** button.
4. In the Monitoring Options section of the window deselect **Message on No Logfile** to prevent error messages from appearing in the Message Browser.

Task 4: Assign/distribute templates and files

In this task you assign/distribute the DB-SPI templates and programs to all cluster nodes capable of running the Oracle packages.

1. Log on to OVO as the OVO administrator
2. To ensure that the DB-SPI messages correctly appear in the Message Browser, in the Node Bank window from the Actions menu, select **Message Browser**.
3. In the Message Browser, select **Restart**.
4. Open the User Bank window, modify the **opc_adm user**, and choose **Responsibilities**. Use the Responsibilities for Operator (opc_adm)window to view the DB-SPI messages.
5. Click the following buttons in the Configure Administrator Message Browser window to select the DB-SPI message groups:
   DBSPI
   Ora_Admin, Inf_Admin, Syb_Admin, or MSS_Admin
   Ora_Conf, Inf_Conf, Syb_Conf, or MSS_Conf
   Ora_Fault, Inf_Fault, Syb_Fault, or MSS_Fault
   Ora_Perf, Inf_Perf, Syb_Perf, or MSS_Perf
6. Click **OK** to add DB-SPI message groups
7. To distribute DB-SPI to the managed nodes, first select the VPO node(s) designated to run DB-SPI
8. From the Actions menu select Agents→Install/Update SW & Config
9. In the Install/Update VPO software and Configuration window check the following check boxes to select these components:
   — Actions
   — Monitors
   — Commands
   (Do NOT select the Templates check box at this point in the installation.)
10. Click the **Nodes in List Requiring Update** button
11. If you did not previously select a node, click the **Get Map Selections** button to list the target node(s)
12. Click the **Force Update** option button

13 To distribute Actions, Commands and Monitors to the managed node(s), click **OK**.

Task 5: Configure database information for each managed node

Prerequisite: Prepare a list of all RDBMS instances on the node that includes keyword information.

1 Open the OVO Node Bank window and highlight the managed node to configure.

2 From the Window menu, select **Application Bank**.

3 Select **DBSPI→Admin→Configure DB Connections** (or drag and drop the managed node onto the application).

4 In the DBSPI Configuration File insert entries for each database instance, referring to the list you prepared (in the prerequisite step).

5 To the following prompt:
`Save configuration to /var/opt/OV/dbspi/local.cfg> [yes]`
Accept the default **Yes** to apply the changes to the node.

(If you want to undo the changes rather then apply them, select N and press Enter at the prompt.)

Task 6: Distribute the configuration to all cluster nodes.

Task 7: Configure Nodes for Cluster Failovers

The DB-SPI Collector/Analyzer suspends collection without reporting errors if the `defaults` file contains the entry:

```
COLLECTION OFF
```

DBSPI provides a utility script to trigger entries in this file and report a change in state to the Message Browser. If the file is non-existent or contains other keywords, collection is enabled.

To turn some collections off while leaving others on: The `defaults` file must be located on the managed node where you want to make the change. If this file is not on the managed node, create it.(For the file location, see Data, Log, and Configuration Files on page 238.)

On the managed node, edit the `defaults` file to contain the line:
<*first_database_instance_name*> OFF
<*second_database_instance_name*> OFF

Example Entries:

```
ora803 OFF
openview ON
SAP1 OFF
```

`COLLECTION OFF` overrides any other entries in the file.

Configure Cluster Software (VERITAS example)

Now that you have configured the Database SPI, the remaining task is to insert entries for the Database SPI data collection within the cluster software. With these entries the Database SPI collector will function in parallel with your cluster software. The example below shows how DB-SPI can be made to work with VERITAS cluster software, ensuring continuous data collection.

For VERITAS Clusters: Edit the VERITAS Cluster script

Use the output of the service registration information to locate the VERITAS cluster script that you will need to edit.

1 Generate the service registration output by entering:
`hareg -q <service_name>`

Example of output:

```
#hareg -q test
    hareg -r test -b "/opt/SUNWcluster/ha/test/" -m
         hareg -r test -b "/opt/SUNWcluster/ha/test/" -m
        START="/opt/SUNWcluster/ha/test/test_svc_start" -t START=60 -m
         STOP="/opt/SUNWcluster/ha/test/test_svc_stop" -t STOP=60 -m
         ABORT="/opt/SUNWcluster/ha/test/test_svc_abort" -m
         START_NET="/opt/SUNWcluster/ha/test/test_svc_start_net" -t
START_NET=60 -m
      STOP_NET="/opt/SUNWcluster/ha/test/test_svc_stop_net" -t
STOP_NET=60 -m        ABORT_NET="/opt/SUNWcluster/ha/test/
test_svc_abort_net" -m
        FM_INIT="/opt/SUNWcluster/ha/test/test_fm_init" -t FM_INIT=60
-m
         FM_START="/opt/SUNWcluster/ha/test/test_fm_start" -t
FM_START=60
-m
        FM_STOP="/opt/SUNWcluster/ha/test/test_fm_stop" -t FM_STOP=60
-m

     FM_CHECK="/opt/SUNWcluster/ha/test/test_fm_check" -t FM_CHECK=60
-h "nfs1" -a 1
```

2 Note the locations for `START_NET` and `STOP_NET`.

3 Edit the `START_NET` script.
In this example, the script is located in: `/opt/SUNWcluster/ha/test/test_svc_start_net`

Add the following line at the end of the function:
`/var/opt/OV/bin/OpC/cmds/dbspicol ON`

4 Edit the `STOP_NET` script.
In this example, the script is located in: `/opt/SUNWcluster/ha/test/test_svc_stop_net`

Add the following line at the end of the function:
`/var/opt/OV/bin/OpC/cmds/dbspicol OFF`

For VERITAS Clusters: Configure the Software

Perform these steps on all nodes that are configured to run the monitored database(s).

1 Find the location of the online/offline scripts by entering:
`$ hares -display <resource> -attribute Type`

2 Use the name in the Value column to locate the directory of the online and offline scripts:

```
$ cd /opt/VRTSVCS/bin/<value>
$ ls -l
```

The example below shows a standard OVO management server installation, which refers to the OVO "openview" Oracle database:

```
$ hares -display vpo -oracle -attribute Type

   #ResourceAttributeSystemValue

   vpo-oracleTypeglobalVPOOracle

   $ cd /opt/VRTSvcs/bin/VPOOracle

   $ ll

   total 326

   -r-xr--r--   1 root     sys        33109 Jan 23 12:52 clean

   -r-xr--r--   1 root     sys        35222 Jan 23 12:53 monitor

   -r-xr--r--   1 root     sys        39563 May 31 16:24 offline

   -r-xr--r--   1 root     sys        39660 May 31 16:25 online

   -rwxr--r--   1 root     sys        17252 Aug  9  2001 VPOOracleAgent
```

Now you must locate the appropriate place in the online and offline scripts to call the `dbspicol` command. No standard layout for these scripts exists, so where you insert this syntax depends on the implementation. Note that if the implementation is monitoring the management server's database, it is likely a standard installation. Additionally, be sure that the same scripts are used for bringing resources of the same type on or off line. The original installer may have implemented several database instances that can be brought online and offline independently with the same script. Such a configuration builds in the logic to determine which database instance's collection must be turned on or off.

In this example (the standard OVO management server installation):

1 Near the end of the online script, locate following lines:

```
#Start Oracle Database.
 ha_ora -stop silent
 ha_ora -start
```

2 Immediately after the lines in the preceding step, insert the lines:

```
   # Turn DBSPI collection on
     [ -x /var/opt/OV/bin/OpC/cmds/dbspicol ]  &&
     /var/opt/OV/bin/OpC/cmds/dbspicol ON openview
```

3 Near the end of the offline script, locate the lines:

```
  # Stop Oracle Database.
  ha_ora -stop
```

4 Immediately before the lines in the preceding step, add following:

```
# Turn DBSPI collection OFF - TT 2002/05/31
  [ -x /var/opt/OV/bin/OpC/cmds/dbspicol ] &&
  /var/opt/OV/bin/OpC/cmds/dbspicol OFF openview
```

8 Error Messages & Troubleshooting

This chapter covers error messages and problem situations you may encounter. Error message text is sorted by number for quick reference. It includes the following:

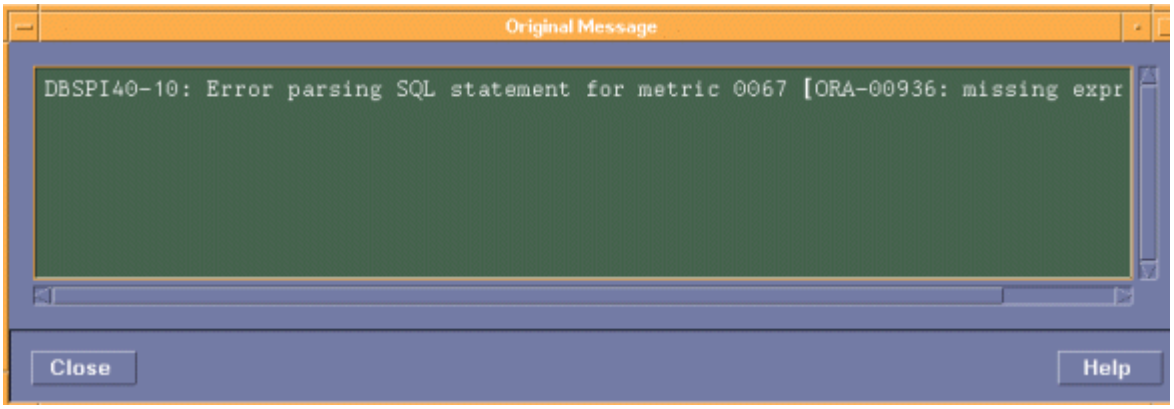
Error Messages

This section provides detailed information about the text contained in the DB-SPI error messages.The following information is provided:

- Error Message Number
- Description
- Severity
- Message Group
- Help Text

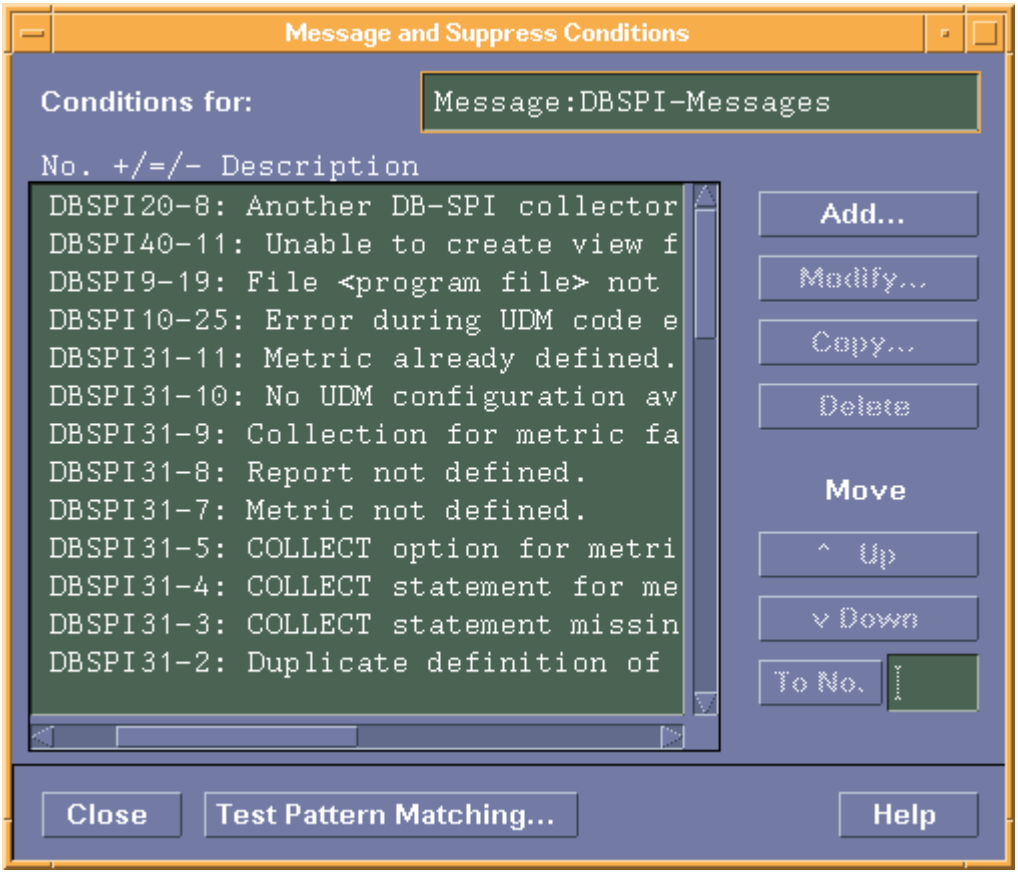
Error messages can be viewed from the OVO Message Browser by opening the desired message and clicking the **Show Original Message** button (see illustration below).

Figure 45 Viewing the Original Error Message Text



These messages are generated by the collector/analyzer and intercepted by the `DBSPI-Messages` template (see illustration below).

Figure 46 Viewing the list of messages in the DBSPI messages template



Eliminating Duplicate Error Messages

The follow procedure is necessary only for OVO 6.x and earlier versions. The suggested modification are unnecessary for the current OVO release.

The OVO template `opcmsg (1|3)` also filters these messages, causing them to appear twice in the OVO Message Browser. To eliminate duplicate messages, follow these steps to add a condition to the `opcmsg(1|3)` template:

1. Open the **Message Source Templates** window.
2. From the *platform group* you are using, select the **opcmsg(1|3)** template and double click it to open the Message and Suppress Conditions window.
3. Press the **Add** button to add a new condition and display the Condition window.
4. Click the **Suppress Matched Condition** option button.
5. In the Message Text field, enter: `DBSPI`
6. Click the **OK** button in the Condition window.

7 Click the **Close** button in the Message and Suppress Conditions window.

Please refer to the OVO Help or the *OVO Administrators Task Guide* for further details on adding conditions to templates.

DBSPI10-1

Description:	Invalid character in range.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: A list of metrics was specified on the command line and the range specified is incorrect. The range must contain numbers, such as 'dbspicao -m 6-8 -p'. An invalid character was entered in the numeric range. Suggested Action: Specify the range correctly.

DBSPI10-2

Description:	Upper bound is lower than lower bound.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: A list of metrics was specified on the command line and the range specified is incorrect. The range must start with the smaller number and end with the larger number. For example, 'dbspicao -m 6-8 -p' is correct but 'dbspicao -m 8-6 -p' is incorrect. Suggested Action: Specify the range correctly.

DBSPI10-3

Description:	Invalid character in range (number expected).
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: A list of metrics was specified on the command line and the range specified is incorrect. The range must contain numbers, such as 'dbspicao -m 6-8 -p'. A non-numeric value was specified such as 'dbspicao -m 6-x -p. Suggested Action: Specify the range correctly.

DBSPI10-21, 22, or 23

Description:	Unable to connect to database.
Severity:	Critical
Message Group:	Ora-Fault
Help Text:	Probable Cause: A connection to the database highlighted in the message failed using the information in the DB-SPI configuration file. This file is automatically created when a ConfigFile template of type dbspi/db_config is deployed to a managed node. The configuration could also have been performed locally via dbspicfg.sh on UNIX or dbspicfg.exe on Windows. The error could be caused by: * The HOME location of the database is incorrect or has changed * The userid or password that is used to connect to the database is incorrect or has changed. * The database configured is no longer on the system * The database configured is not in proper running order. The database could have been shutdown for maintenance, or another more serious, unplanned failure may have occurred. * Other incorrect configuration information was entered

Description:	Unable to connect to database.

	Suggested Action: Use the database name and error in the message to determine which database is failing and why. If the database is up and running, then closely examine the configuration using the ConfigFile template editor and look for errors.
	The Oracle two errors that can occur in relation to directory or file permissions are: (1) ORA-01034: ORACLE not available ; ORA-07318: smsget: open error when opening sgadef.dbf file. ; HP-UX Error: 13: Permission denied (2) ORA-12546: TNS:permission denied.
	If either of these Oracle errors occurs, check the following: |
	* Ensure that
	`$ORACLE_HOME and $ORACLE_HOME/bin`
	have the permissions set to: `drwxr-xr-x`
	* Ensure that the $ORACLE_HOME/bin/oracle file has the permissions set to: -rwsr-s--x
	If not set as above, enter this command:
	* `chmod 6755 $ORACLE_HOME/bin/oracle`
	If you do not want DB-SPI to collect data for a prolonged period, or you are planning a shutdown and you want to avoid this message, it is recommended you turn off the DBSPI collection. To do so, use the DB-SPI tools:
	Stop Monitoring- Temporarily disables metric data collection and alarming on the node where this tool was started.
	Start Monitoring- Re-enables metric data collection and alarming.
	To turn monitoring ON and OFF from a command line use the 'dbspicol' tool:
	UNIX: /<*Commands_directory*>`/dbspicol -v OFF | ON`
	NT: \<*Commands_directory*>`\dbspicol -v OFF | ON`
	For example, on HP-UX:
	`/var/opt/OV/bin/OpC/vpwin/cmds/dbspicol -v OFF`
	`/var/opt/OV/bin/OpC/vpwin/cmds/dbspicol -v ON`
	To turn monitoring OFF for a single instance, modify the 'defaults' file by adding the instance name followed by OFF. The turn monitoring back on, remove the line or change the OFF to ON, for example: SAPdb OFF
	NOTE - If the connections continue to fail, this message will appear only once every 2 hours.

DBSPI10-24

Description:	Connect to DB timed out after seconds. Please check database.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: The database that DB-SPI is connecting to timed out after the number of seconds in the message. The database is likely not responding. Suggested Action: Attempt to connect to the database using the same parameters as those configured using SQL*Plus. You may have to shut down and start up the database.

DBSPI10-25

Description:	Error during UDM code execution for metric.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: An error occurred executing a UDM. Suggested Action: Use the UDM metric number and the error returned from the RDBMS to determine the cause of the error. It is likely a syntax or logic error in the PL/SQL used to define the UDM. Attempt to use the message details to fix the error. If necessary, you may have to take the UDM PL/SQL and test it in SQL*Plus to determine the cause of the error.

DBSPI10-27

Description:	Oracle Listener process is not running.
Severity:	Minor
Message Group:	DBSPI

Description:	Oracle Listener process is not running.
Help Text:	Probable Cause: The script dbspi_lsnr was run from the template DBSPI-Ora-Listener. This script discovered that the process 'tnslsnr' is not running by evaluating the results of the following command: `ps -fu oracle | grep -i tnslsnr` Suggested Action: Restart the Oracle Listener process via 'lsnrctl start'.

DBSPI10-28

Description:	Unable to check Oracle Listener; DBSPI is not configured.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: The script `dbspi_lsnr` was run from the template DBSPI-Ora-Listener. This script determined that DB-SPI has not yet been configured. Suggested Action: Use the 'DBSPI-Config' application in the 'DBSPI Admin' application group to configure 1 or more Oracle instances.

DBSPI10-29

Description:	One or more Oracle network services are not configured.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: The script dbspi_lsnr was run from the template DBSPI-Ora-Listener. This script discovered that one or more network services are not configured by examining the results of the following command: `lsnrctl status | grep TNS` Suggested Action: Stop and restart the Oracle Listener via 'lsnrctl stop' followed by 'lsnrctl start'.

DBSPI10-30

Description:	Cannot connect to the Oracle Listener using *<user/pass>*.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: The template DBSPI-Ora-Listener-Connect is being used to check the SQL*Net connections to all or specific databases configured for DB-SPI. This template runs a special DB-SPI collector/analyzer option '-n': dbspicao -c DBSPI-Ora-Listener-Connect -n The -n option will attempt to connect to each database (or only one if the -i option is used) using a tnsnames.ora entry with the exact same name as the Oracle instance name. For example, if you had the following Oracle instances configured in DB-SPI: ORACLE HOME '/opt/oracle/8.0.5' DATABASE sales CONNECT 'hp_dbspi/hp_dbspi' DATABASE finance CONNECT 'hp_dbspi/hp_dbspi' The collector/analyzer will attempt to connect to the following instances: hp_dbspi/hp_dbspi@sales hp_dbspi/hp_dbspi@finance If this connection fails, message DBSPI10-30 is generated.

Description:	Cannot connect to the Oracle Listener using *<user/pass>*.
	Suggested Action: In order for this connection check to work, you MUST have one tnsnames.ora entry matching the instance configured for each instance name (or if you are using a different facility to register the remote name, then modify that facility). In the above example, you would need the following two entries in tnsnames.ora for the connection check to work: sales= (DESCRIPTION= (ADDRESS_LIST= (ADDRESS = (PROTOCOL = IPC)(KEY =sales)) (ADDRESS = (COMMUNITY=OPENVIEW_COMMUNITY) (PROTOCOL=TCP) (HOST=vwbug) (PORT=1521))) (CONNECT_DATA= (SID=sales))) finance= (DESCRIPTION= (ADDRESS_LIST= (ADDRESS = (PROTOCOL = IPC)(KEY =finance)) (ADDRESS = (COMMUNITY=OPENVIEW_COMMUNITY) (PROTOCOL=TCP) (HOST=vwbug) (PORT=1521))) (CONNECT_DATA= (SID=finance)))

Description:	Cannot connect to the Oracle Listener using *<user/pass>*.
	Suggested Action (contd): The tnsnames.ora entry MUST match the instance name exactly. If you do not intend to perform this check, do not assign the DBSPI-Ora-Listener-Connect template to the node. If you intend to check only specific Oracle instances, copy the template and add the '-i' option to the command line. For example, you would copy the DBSPI-Ora-Listener-Connect template to a different template for each instance that you want to check and then modify the command line similar to the following: dbspicao -c DBSPI-Ora-Listener-Connect -n -i sales Create a separate template for each instance that you want to check and assign these policies to the appropriate node. If the above configuration was performed and the error still occurs, then restart the Oracle Listener.

DBSPI10-31

Description:	Oracle 7.3.x is not supported on HP-UX 11.x with OVO 5.x or 6.x.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: A database was configured on an HP-UX 11.x system running an OVO agent version 5.x. Starting with version 5, DB-SPI cannot support Oracle 7.x in this configuration. Only 8.x can be supported on HP-UX 11.x running OVO agent version 5 or higher. This is due to an incompatibility in libraries with Oracle and OVO. Suggested Action: * Upgrade to Oracle 8.x or higher. * Monitor the database remotely from an OVO 4.x node. If these actions are not possible, then the database cannot be monitored.

DBSPI10-51

Description:	An internal IT/O error has occurred.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: An internal OVO error has occurred. Suggested Action: Verify a correct installation of IT/O and that OVO agents are running (opcagt -status).

DBSPI10-81

Description:	Unable to gain lock file. File already in use by another process.
Severity:	Major
Message Group:	DBSPI
Help Text:	Probable Cause: The collector/analyzer was unable to access a shared file. Another collector/analyzer already has the file in use. Suggested Action: If the modification date on the reported file is greater than one hour old, it is most likely that the process holding this file has aborted. In this case remove the file using the rm command.

DBSPI19-24

Description:	Old DBSPI files have been found on this node.
Severity:	Warning
Message Group:	DBSPI

Description:	Old DBSPI files have been found on this node.
Help Text:	Probable Cause: DB-SPI files were previously distributed to this node from OVO or VPO and the node is now managed by OVO for Windows. Suggested Action: Delete all files that start with 'dbspi' from the following directories: Operating SystemDirectories ========== === Windows`\usr\OV\bin\OpC\intel\cmds` \usr\OV\bin\OpC\intel\monitor \usr\OV\bin\OpC\intel\actions \usr\OV\bin\OpC\cmds \usr\OV\bin\OpC\monitor \usr\OV\bin\OpC\actions HP-UX, Linux, Solaris, or Tru64 `/var/opt/OV/bin/OpC/cmds` /var/opt/OV/bin/OpC/monitor /var/opt/OV/bin/OpC/actions AIX`/var/lpp/OV/OpC/cmds` /var/lpp/OV/OpC/monitor /var/lpp/OV/OpC/actions

DBSPI19-25

Description:	Old DBSPI templates/policies have been distributed to this node.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: DB-SPI templates/policies were previously distributed to this node from OVO or VPO and the node is now managed by VPW. Suggested Action: Run the following command to see if any old DB-SPI templates/policies have been deployed to the node: UNIX : /opt/OV/bin/OpC/utils/opcdcode /var/opt/OV/conf/OpC/monitor | grep DBSPI Windows : %windir%\Hewlett-Packard\OVEnterprise\Agent\NgSB\{790C06B4-844E-11D2-972B-080009EF8C2A}\bin\OpC\utils\opcdcode \usr\OV\conf\OpC\monitor >c:\temp\monitor.tx Then open c:\temp\monitor.txt in an editor and scan for DBSPI. If these templates/policies do exist, they MUST be deleted. To remove these and all other old templates/policies, delete the following files: Operating SystemFiles To Delete ========== HP-UX, Linux, Solaris, or Tru64 /var/opt/OV/conf/OpC/monitor /var/opt/OV/conf/OpC/le /var/opt/OV/conf/OpC/msgi AIX/var/lpp/OV/conf/OpC/monitor /var/lpp/OV/conf/OpC/le /var/lpp/OV/conf/OpC/msgi Windows \usr\OV\conf\OpC\monitor \usr\OV\conf\OpC\le \usr\OV\conf\OpC\msgi If files remain in this directory, remove them by entering: rm -rf /var/opt/OV/conf/OpC/DBSPI After removing the old template files, restart the agent using: opcagt -start

DBSPI20-1

Description:	The configuration file does not exist. Please configure this node.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: The monitored node has not been configured for DB-SPI, or the node to which the DB-SPI policies were distributed has no databases to monitor. Suggested Action: Follow the instructions in the manual to create a DB-SPI configuration file and deploy it to the managed node. If you continue to have a problem, run the application "Check Connections." to check the configuration and examine the output from this tool in an attempt to determine what went wrong. If you do not intend to monitor databases on this system, modify the template assignment to exclude the node from DBSPI template distribution.

DBSPI20-2

Description:	Metric has no report defined.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: An attempt was made to generate a report using the DB-SPI collector/analyzer (dbspicax) from a UNIX command line or from OVO as an operator or automatic action for a metric that does not have a report defined. For example: /var/opt/OV/bin/OpC/monitor/dbspicao -r 1 -m 2 Suggested Action: It is likely that an attempt was made to generate a report for a different metric and a typing mistake was made. Correct the call to the collector/ analyzer to specify the correct metric number. The list of metrics that support a report is documented in the *Reference* manual (see the HP OpenView Smart Plug-in for <*database_type*>: Reference.

DBSPI20-3

Description:	Collection/Analysis has been turned ON/OFF
Severity:	Warning
Message Group:	DBSPI

Help Text:	Probable Cause:
	The collection/analysis has been turned ON or OFF for this managed node. This was performed via the Application Bank applications Start Monitoring or Stop Monitoring or through the command line:
	UNIX:/<Commands_directory>/dbspicol -v OFF | ON
	NT:\<Commands_directory>\dbspicol -v OFF | ON
	For example, on HP-UX:
	`/var/opt/OV/bin/OpC/cmds/dbspicol -v OFF`
	`/var/opt/OV/bin/OpC/cmds/dbspicol -v ON`
	Suggested Action:
	If the collection/analysis was turned off inadvertently, turn it back on immediately. If it was turned off on deliberately, turn it back on later when desired.

DBSPI20-5

Description:	DB-SPI DSI logfiles need to be converted.
Severity:	Major
Message Group:	DBSPI

Description:	DB-SPI DSI logfiles need to be converted.
Help Text:	Probable Cause: Starting with A.03.20, the DSI logfile format for the Oracle MeasureWare DSI logfile has changed. This change requires that the DB-SPI DSI logfiles be moved and rebuilt. Until these files are moved and rebuilt as described below, DB-SPI data cannot be sent to MeasureWare. Suggested Action: In order to update the DSI logfiles to the new A.03.20 format, it is necessary to first shutdown the MeasureWare servers, move and rebuild the logfiles, then restart the MeasureWare servers. If you are ready to do this now, perform the Operator Action associated with this message. The Operator Action executes the following script: Suggested Action: /var/opt/OV/bin/OpC/cmds/dbspi_mw_con (for OVO) Which does the following: (1) Stops MeasureWare Agent servers (mwa stop servers). (2) Executes the dbspi_mw_int - script, which moves the existing DB-SPI Oracle DSI logfiles to /var/opt/OV/dbspi/DSI.OLD/oracle/ and rebuilds new logfiles in /var/opt/OV/dbspi/dsi/oracle. (3) Executes the dbspi_mw_int udm - script, which rebuilds User Defined Metric logfiles in /var/opt/OV/dbspi/dsi/oracle/udm. (4) Starts MeasureWare Agent servers (mwa start servers).
	Suggested Action (contd.): Or you can execute the script at a later time from the command line prompt. You can still view the old DSI logfiles. The appropriate entries are updated in `/var/opt/perf/perflbd.rc`. The entries for the old log files have the format: DATASOURCE=ORA_ora733 LOGFILE=/var/opt/OV/dbspi/DSI.OLD/oracle/ora733.log The entries for the new logfiles have the format: DATASOURCE=ORADB_kimball_ora733 LOGFILE=/var/opt/OV/dbspi/dsi/oracle/ora733.log If you no longer need access to the old MeasureWare logfiles, delete the entries in /var/opt/perf/perflbd.rc that point to the 'DSI.OLD' directory.

DBSPI30-1

Description:	Syntax error.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: A syntax error was encountered when editing: * The DB-SPI configuration files using Configure DB Connection or the command line dbspicfg. OR * User Defined Commands using the Configure UDM application or the command line dbspi_pcheck. The first parameter is the line number, the second parameter is a more specific error message and the third parameter is the exact token that caused the error. Suggested Action: Correct the syntax error and try again.

DBSPI30-2

Description:	No instance defined for <*HOME*> - ignored.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: No 'DATABASE... CONNECT...' clause was specified after a HOME clause. The HOME clause will be ignored. Suggested Action: Either remove the HOME clause or add a 'DATABASE...CONNECT...' clause under the HOME clause.

DBSPI30-3

Description:	<Parm> already defined earlier - ignoring this definition.
Severity:	Warning

Description:	<Parm> already defined earlier - ignoring this definition.
Message Group:	DBSPI
Help Text:	Probable Cause: A duplicate definition was encountered and ignored. Suggested Action: Remove the duplicate definition.

DBSPI30-4

Description:	Filter already defined earlier - ignoring this definition.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: The same filter definition was included twice for one metric for a database definition. Suggested Action: Remove the duplicate filter.

DBSPI31-1

Description:	Report already defined.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, the same REPORT number was used more than once. For example: METRIC 0701 COLLECT MW OVO '....' REPORT 1 INCLUDE 'Ora0701-1.sql' REPORT 1 INCLUDE 'Ora0701-2.sql' Suggested Action: Fix the configuration to eliminate the duplicate entry.

DBSPI31-2

Description:	Duplicate definition of database.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, more then one definition section for a specific database type. For example, sections for ORACLE appear more than once. Suggested Action: Specify only one section for each database. Then eliminate or merge the two sections.

DBSPI31-3

Description:	COLLECT statement missing for metric.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, a metric specification does not contain a COLLECT statement. For example: METRIC 0701 REPORT 1 INCLUDE 'Ora0701-1.sql' is not allowed. You must specify a COLLECT statement. Suggested Action: If you want the metric strictly for reporting purposes, specify a dummy COLLECT clause and simply never execute the metric. For example: METRIC 0701 COLLECT 'Ignore This' REPORT 1 INCLUDE 'Ora0701-1.sql'

DBSPI31-4

Description:	COLLECT statement for metric already defined.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, more than one COLLECT statement was specified. For example: METRIC 0701 COLLECT MW OVO '....' COLLECT MW '....' Suggested Action: Fix the configuration to eliminate the duplicate entry.

DBSPI31-5

Description:	COLLECT option for metric already defined - ignored.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, more than one of the same options was specified in the COLLECT statement. For example: METRIC 0701 COLLECT MW OVO MW '....' Suggested Action: Even though the entry is ignored, the duplicate entry should be eliminated.

DBSPI31-7

Description:	Metric not defined.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: An attempt has been made to execute a User Defined Metric via the DBSPI collector analyzer that has not been configured on the managed node. For example: dbspicao -m 755 -p -v In this example, if metric 755 has not been defined using the UDM Config and distributed to the managed nodes, then DBSPI31-7 results. Suggested Action: Use the UDM Config application to configure the metric number being executed or modify the command line to specify only metrics that have been properly defined. It is also possible that you have not yet distributed the configuration to the managed nodes. You need to distribute the UDM configuration via the OVO GUI using Action | Agents | Install/Update SW & Config as described in the documentation before the UDM becomes active on the managed node.

DBSPI31-8

Description:	Report not defined.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: An attempt was made to execute a User Defined Metric report and the REPORT number being specified has not been defined. For example: `dbspicao -m 701 -r 2` And the UDM configuration for 701 looks like this: METRIC 0701 COLLECT MW OVO '....' REPORT 1 '...' Suggested Action: Correct the command to specify only reports that are defined or add the REPORT definition to the configuration using the UDM Config application and distribute Monitors via the OVO GUI using Action | Agents | Install/Update SW & Config as described in the documentation.

DBSPI31-9

Description:	Collection for metric failed (`dbspi_error=...`)
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: An error occurred while executing a User Defined Metric. The error message returned (following dbspi_error=) is one that is returned via user defined logic within the UDM. Suggested Action: Examine the error and determine the cause from the error. Correct the error or UDM logic and try the UDM again.

DBSPI31-10

Description:	No UDM configuration available (`config file` installed?).
Severity:	Major
Message Group:	DBSPI
Help Text:	Probable Cause: An attempt has been made to execute a User Defined Metric via the DBSPI collector analyzer and a User Defined Metric configuration has not been setup. For example: `dbspicao -m 755 -p -v` Suggested Action: Use the UDM Config application to configure the UDM's for your systems and then distribute Monitors via the OVO GUI using Action | Agents | Install/ Update SW & Config as described in the documentation.

DBSPI31-11

Description:	Metric already defined.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: Configuring User Defined Metrics, the same METRIC number was used more than once. For example: METRIC 0701 COLLECT MW OVO 'PL/SQL for Metric 0701....' REPORT 1 INCLUDE 'Ora0701.sql' METRIC 0701 COLLECT MW OVO "PL/SQL for Metric 0702...." REPORT 1 INCLUDE 'Ora0702.sql' Suggested Action: Fix the configuration to eliminate the duplicate metric number.

DBSPI40-11

Description:	Unable to create view for table <table> for metric <metric>.
Severity:	Minor
Message Group:	DBSPI
Help Text:	Probable Cause: DBSPI-0003 or DBSPI-0006 metric could not create the view necessary to run the automatic action tablespace report. This occurred because the user connected to Oracle does not have the necessary privileges to create the view that is needed for the report. Suggested Action: Follow the instructions in Chapter 2 (OVO Enterprise) or Chapter 3 (OVO Special Edition) to grant the appropriate privileges to the user that is connecting to Oracle for DB-SPI.

DBSPI40-13

Description:	Connection attempt to server <*sybase_server*> with user <*user_name*> failed.
Severity:	Critical
Message Group:	DBSPI

Help Text:	Probable Cause:
	This error occurs when DB-SPI is unable to connect to the Sybase server.
	Suggested Action:
	Check the availability of the server to make sure that it is up and running. Try executing the following commands on the managed node that encountered the error (as `root`):
	`/var/opt/OV/bin/OpC/monitor/dbspicas -d -v`
	Or use the 'DBSPI Connect' application in the 'DBSPI Admin' application group.
	If this fails, then check the configuration by executing:
	`/var/opt/OV/bin/OpC/cmds/dbspicfg.sh`
	Or use the 'Configure DB Connections' application in the 'DBSPI Admin' application group.
	Check the configuration. When saving the configuration, the script will again check the connections.

DBSPI40-14

Description:	The call to function %s in %s failed.
Severity:	Warning
Message Group:	DBSPI
Help Text:	Probable Cause: The error could be caused by a variety of different calls to the Sybase DBMS. However, one error in particular is likely to occur and has two easy workarounds. If the following error appears in the OVO message browser or in the dbspierror file, then please follow one of the suggested actions described below: 05/04/99 14:10:21 ERROR dbspicas(2292) MUDBUG [cola:syb_util.c:767]: DBSPI40-14: The call to function init_context in ovam_sql_connect_db2 failed. This problem is caused by the START_LANG localization environment variable in the file `/etc/rc.config.d/opcagt`. For OVO 5.0, this variable will be set to the following value: START_LANG=C.iso88591 The problem is that the default Sybase configuration does not know about this language.
	Suggested Action: There are 2 possible solutions to the problem. The easiest is to modify the file /etc/rc.config.d/opcagt and change the START_LANG line as shown: START_LANG=C After making this change, restart the OVO agents: `/opt/OV/bin/OpC/opcagt -kill` /opt/OV/bin/OpC/opcagt -start If the C.iso88591 language is needed, add the following entries to the file $SYBASE/locales/locales.dat For HP-UX: [hp ux] locale = C.iso88591, us_english, roman8 For Sun Solaris: [sun] locale = C.iso88591, us_english, roman8 Place this entry adjacent to the entry: locale = C, us_english, roman8 After making this change, the error should cease.

DBSPI40-15

Description:	Unable to load dbspi stored procedures on server <sybase_server>.
Severity:	Critical
Message Group:	DBSPI
Help Text:	Probable Cause: This error indicates a failure in executing the script /var/opt/OV/bin/OpC/monitor/dbspi_sybsp This script is necessary to create stored procedures which are used by DB-SPI. Suggested Action: Try executing the script by hand using Sybase's ISQL tool to determine if there is some sort of problem with the Sybase database. This could indicate a security or permission problem with the user that is connecting to the server or a database space or transaction log full problem with `sybsystemprocs` database. Once the script is successfully executed by hand using ISQL, it will not need to be executed again by DBSPI.

DBSPI40-16

Description:	Sysprofile name '%s' not found. Unable to calculate metric.
Severity:	Minor
Message Group:	DBSPI

Description:	Sysprofile name '%s' not found. Unable to calculate metric.
Help Text:	Probable Cause: This error indicates a failure in finding sysprofile entries needed to calculate a metric. Suggested Action: Try connecting to the sysmaster database using the configured DBSPI userid and running the following query: select name, value from sysprofile; Look for the name listed in the error message. Grant select permissions to the DBSPI user-id, if needed. If SMI tables (e.g., sysprofile) are not correctly built, they may need to be re-built. If the above query runs successfully, but the name that DBSPI is looking for does not appear in the output, you may need to disable the metric and contact HP Support..

All Remaining Errors

Description:	All remaining errors.
Severity:	Warning
Message Group:	DBSPI

Description:	All remaining errors.
Help Text:	Probable Cause: An unexpected DB-SPI error occurred. This could be caused by an internal error (code failure) or by external forces such as unavailable disk space, operating system problems, database problems, etc. Suggested Action: The error message could have one or more of the following error types included in the message: * File system error * Operating system error * Database error Use these errors and the appropriate documentation (HP-UX manuals, Oracle manuals, etc.) in an attempt to determine if system problems of some type exist (unavailable disk space, low on system memory, Oracle problems, etc.). If no apparent system problem exists, an internal DB-SPI code failure is possible. To help determine the failure, follow these steps: (1) Turn tracing on by running the DB-SPI Admin application: Trace On. (2) Try to reproduce the problem and record all the steps necessary as you proceed. (3) Be sure to turn off tracing by runnning the DB-SPI Admin application: Trace Off. (4) Run "Verify Deployment" and save the output results. (5) Call HP Support and provide the support representative with the following: A. The results of "Verify Deployment" B. The steps and other information on reproducing the problem C. The trace file

Troubleshooting Special Situations

This section describes a troubleshooting aid and known issues. It includes the following sections:

- Turning on Tracing
- Using the Self-Healing Info Application
- UDM Configuration Issues

Turning on Tracing

If a problem occurs and you are asked to turn on tracing by the DB-SPI support team, do the following:

1. Open the Node Bank window and select the node for which you want tracing turned on.
2. Open the Application Bank window.
3. Double-click **DB-SPI→Admin**.
4. Double-click **Trace On**.

Using Commands for Specific Tracing

When you need to look only at certain metrics, you can selectively log data for them only as follows:

Table 43 Collection and Tracing commands, with parameters

Function	Command
To re-enable performance metric data collection and notification	`dbspicol ON`
To temporarily disables performance metric collection and alert notification	`dbspicol OFF`
To re-enable performance metric data collection and notification for a specific database instance	`dbspicol ON <instance>`
To temporarily disable performance metric collection and alert notification for a specific database instance	`dbspicol OFF <instance>`
To turn on DB-SPI tracing	`dbspicol TRACE`
To turn off DB-SPI tracing	`dbspicol TRACEOFF`
To turns on DB-SPI tracing for a specific instance	`dbspicol TRACE <instance>`

Table 43 Collection and Tracing commands, with parameters

Function	Command
To turn off DB-SPI tracing for a specific instance	`dbspicol TRACEOFF <instance>`
To turn tracing on for specific metrics **Note**: The example is correct; you do not use the word "`TRACE`" in the syntax.	`dbspicol [oracle|sybase|informix|mssql] <metric#,metric#>` Example: `dbspicol oracle 8,16` Example: `dbspicol informix 2, 15-25` Example: `dbspicol sybase 35, 235` Example: `dbspicol mssql 2, 3`

To turn tracing off for specific metrics (to stop the tracing you previously initiated)
Note: The example is correct; you do not use the word "`TRACEOFF`" in the syntax.
	`dbspicol oracle`
	`dbspicol informix`
	`dbspicol sybase`
	`dbspicol mssql`

Using the Self-Healing Info Application

- **Preferred troubleshooting method:** Troubleshooting information is greatly enhanced with cross-referencing capabilities available through the HP support site: **http://support.openview.hp.com/self_healing.jsp**. To access this information and the search capabilities available on the site, you must first download and configure the **Self-Healing Services** client software, an added benefit of your HP support contract. A link is then automatically set up to the Self-Healing Services web site, where your SPI problem can be matched to similar SPI problems/solutions for increased troubleshooting effectiveness. Please see the *SPI DVD Installation Guide* section on Self-Healing Services for information on how to download, configure, and use the software and its automatic link to the HP web site support service.
- **Alternative troubleshooting method:** If *Self-Healing Services* is not installed/ configured on the node, you can use the *Self-Healing Info* application. This application also gathers SPI troubleshooting data and stores it in a file that you submit to HP support for assistance.

Prerequisite: Prior to using the Self-Healing Info application: (1) turn on tracing (covered in the previous section), (2) reproduce the problem, and (3) complete these steps:

1. Open the Node Bank window and locate the node on which you want to run the Self-Healing Info application.
2. Open the Application Bank window and double-click **DB-SPI→Admin**.
3. Drag and drop the node onto the **Self-Healing Info** application.
4. In the message that appears, note where the compressed file will be stored.

Depending on a Windows setting, the file may be a hidden file on some managed nodes. If you do not see the file, open Windows Explorer and from the Tools menu select **Folder Options...→View** tabbed page. Under Hidden Files and Folders, select **Show Files and Folders**.

5. In your call to HP support, you can then send the file, as your support representative directs you, as part of your support case.

UDM Configuration Issues

The `dbspiudm.cfg` file, which is installed in

/var/opt/OV/share/databases/OpC/mgd_node/customer
/<*application*>/monitor

is NOT removed when DBSPI is removed from the management server. UDM master configuration is located on the management server in `/opt/OV/DBSPI/conf/udm.cfg`. This file is copied to the path/directory:

```
/var/opt/OV/share/databases/OpC/mgd_node/customer/
   <application>/monitor
```

directory by the UDM config application.

Since `/opt/OV/DBSPI/conf/udm.cfg` may contain extensive customizations, it is a good practice to include this file in the system backup process. Please note that this file is removed by the DB-SPI de-install process (`swremove DBSPI`).

Start DB Application fails on Sybase 15.0

The `Start DB` application fails for Sybase 15.0. The application generates the following error:

```
Starting Database Server: DSPIHP4 while executing

*** Executing '/opt/sybase15/ASE-15_0/install/startserver -f /opt/
sybase15/ASE-15_0/install/RUN_DSPIHP4' for Sybase Server: DSPIHP4

/usr/lib/pa20_64/dld.sl: Unable to find library 'libsbgse2.sl'.
/opt/sybase15/ASE-15_0/install/RUN_DSPIHP4[10]: 19118 Killed
```

To prevent this problem from occurring, you must add the location of Sybase library files to the environment variable, SHLIB_PATH.

Example:

```
# SHLIB_PATH=/opt/sybase15/OCS-15_0/lib3p/libsbgse2.sl
# export SHLIB_PATH=/opt/sybase15/OCS-15_0/lib3p/libsbgse2.sl
```

9 Non-English Environments

This chapter offers an overview of the Database SPI limitations and issues. It also provides guidelines for successful use.

General Limitations

In general, the Database Smart Plug-in is compatible with any database character set because the monitored database information is stored in ASCII text, even though some limitations exist. To clarify the restrictions/requirements of which you should be aware, the following topics are covered:

Limitations with Non-ASCII Character Sets

In general, a DBMS handles output and translation issues as follows:

- ASCII character sets are supported.
- Non-ASCII characters must be converted from the internal character set to the client character set.

In general the Database SPI handles input and translation issues as follows:

- The Database SPI functionality is concerned only with the characters it processes. For example, a Shift JIS database would most likely contain many double-byte characters, but because the Database SPI deals mainly with single-byte numeric values, double-byte character recognition is not an issue.
- Multi-byte character sets, such as Shift JIS, are supported only when the characters used by the Database SPI have been appropriately translated into single-byte characters. Refer to *DB-SPI Oracle Configuration*, 9-222 of this chapter for more information on character set mapping and the Database SPI configuration.

Actions required for customers with non-ASCII databases are summarized in the following table. Discussions of each area follow.

Table 44 Required Actions for Non-ASCII Databases

Area	Action
configuration files	• Use only single-byte characters • Use ASCII user/password • For any non-ASCII characters, check for correct translation
directory and file names	• Disable the "File System Monitor" if local file system names include multi-byte characters • Use Oracle NLS for non-ASCII character translation of data file and archive log directory and file names
character columns used in metrics & reports	• Disable or filter specific metrics which may be affected by translation problems with non-ASCII tablespace names, segment names, schema names, data file names and user names.
DB-SPI NLS configuration	• Configure Oracle NLS environment variables in the `opcagt` start-up script and the root and Oracle user profiles.

Configuration Files

The DB-SPI configuration files (for example, files for Configure DB Connections and user-defined metrics) have the following restrictions:

- Only single-byte characters are allowed in DB-SPI configuration files
- Database name must be in ASCII text
- User-ID/Password must be in ASCII text
- Non-ASCII characters are allowed only in the quoted fields

Characters used in filters and user-defined metrics will be translated to the database character-set by the DBMS. Use only characters that can be mapped to the database character set. Characters output for user-defined metric reports is translated to the managed node and management server character sets. Avoid reporting any columns that may contain data that will not be handled properly.

Oracle Directory and File Names

Only single-byte characters are supported in directories and file names used by the DB-SPI. Single-byte non-ASCII characters can also cause incorrect results due to potential character translation problems. Use the appropriate Oracle NLS configuration to handle character translations, if non-ASCII characters occur in relevant directory and file names.

Table 45 Directory/File Name Specifications for Non-ASCII Characters

Directory/File Name	Specifications
File System Monitor	Do not use the File System Monitor unless all local files system names are single-byte.
Archive Logs	Non-ASCII characters in archive log paths will be translated using Oracle NLS.
Dump Destinations	Background, user and core dump destinations will be translated using Oracle NLS
Data Files	Non-ASCII characters in data file paths will be translated using Oracle NLS.

Character Columns Used in Metrics and Reports

In many cases the *Report conversion* is the only issue. In such cases the metric calculations are correct, but the metric reports contain problems. Report problems could be due to the report containing characters (such as multi-byte) that OVO and Oracle NLS cannot translate correctly. Multi-byte characters, which are not supported, can cause other more significant issues than those listed here.

Table 46 Issues with Non-ASCII Data by Metric

Metric	DB Object	Issues
1		None
2		None
4	User name Tablespace name (default and temporary)	Report conversion
5	Segment name and owner segments in system)	Report conversion
6	Tablespace name Data file name	Tablespace name characters must convert successfully for both the report and the OVO message. Do not use this metric if tablespace names may contain multi-byte characters.
7	Tablespace name	Report conversion
8	Tablespace name	Tablespace name characters must convert successfully for both the report and the OVO message. Data file name characters must convert correctly for the report.

Table 46 Issues with Non-ASCII Data by Metric

Metric	DB Object	Issues
9,11	Data file name	Tablespace name characters must convert successfully for both the report and the OVO message.
16	Segment name and owner Tablespace name Data file name	Do not use this metric if any of these values might contain multi-byte characters. Characters that do not map successfully to the managed node characters set may cause incorrect results in the metric value and the report.
17,18	Segment name and owner Tablespace name	Do not use this metric if any of these values might contain multi-byte characters. Characters that do not map successfully to the managed node characters set may cause incorrect results in the metric value and the report.
19-24,26-28		None
29	User name Index or table partition name Package or package body name or Trigger name	Report conversion
30		None
31	User name	
32-35		None
37	User name	Report conversion
38-41		None
42	Table name and owner Index name and owner	Report conversion
45		None
47	Table name and owner	Report conversion
48-52, 54		None
56	Log archive dest	Log archive dest characters must convert successfully for both the metric value and the report.

Table 46 Issues with Non-ASCII Data by Metric

Metric	DB Object	Issues
57	Log archive dest	Report conversion
58	Log archive dest	Log archive dest characters must convert successfully for both the metric value and the report.
59-61		None
62	Background dump dest	Background dump dest characters must convert successfully for the metric.
63	Background dump dest User dump dest Core dump dest	Background, user and core dump dest characters must convert successfully for the metric and the report.
64	User dump dest	User dump dest characters must convert successfully for the metric.
65	Core dump dest	All dump dest characters must convert successfully for the metric and the report.
66	Background dump dest	Background dump dest characters must convert successfully for the metric and the report.
67,68,69	Rollback segment name	Report conversion
75,77		None
78	Index partition name and owner Table partition name and owner Package name and owner Package body name and owner Trigger name and owner	Report conversion
79	Trigger name and owner Table name and owner	Report conversion
80	Constraint name and owner	Report conversion
81	Snapshot name and owner Snapshot table name	Report conversion
82,83,85,87,89		None

DB-SPI Oracle Configuration

The following environment configurations affect the character conversions needed to support non-ASCII characters:

- management server database character set and GUI character set
- managed node character set
- Database SPI character set
- managed Oracle database character set

A managed node's and a management server's character sets can differ because of the individual settings available on the system. The Database SPI communicates with the management server through the managed node processes. For proper translation to occur, the Database SPI sends messages and reports to the managed node processes using the managed node character set. Much of the information in the Database SPI messages and reports is collected from the Oracle database. The conversion of characters from the Oracle database character set to the managed node character set is done by the Oracle DBMS using Oracle NLS features.

Oracle NLS allows specification of a character set when a database is created. For databases created with character sets other than US7ASCII, character conversion may be necessary in order to map the database character set to the managed node character set. You can accomplish this by modifying environment variables in three places as described in the table that follows.

Table 47 Database SPI set up files for establishing node environment variables (in UX); maps the node character set to the Oracle character set

Environment Variable	Description
`/etc/rc.config.d/opcagt`	This start-up script sets the environment before the OpC agents are started. This environment affects OpC messages and annotations.
`~oracle/.profile`	The DB-SPI Oracle applications in the Application Bank (for example, SQL*Plus) use the.profile of the oracle user.
`~root/.profile`	The DB-SPI Oracle Reports in the Application Bank use the .profile of root.

The environment variables needed for Oracle are described in the following table.

Table 48 Environment Variables for Mapping DB Characters to Oracle

Oracle Environment Variable	Description
NLS_LANG	Set to the Oracle language, territory and character set that match the managed node character set. For example: AMERICAN_AMERICA.WE8ISO8859P1 AMERICAN_AMERICA.WE8ROMAN8 AMERICAN_AMERICA.WE8MSWIN1252 JAPANESE_JAPAN.JA16SJIS JAPANESE_JAPAN.JA16EUC
Set one of the following: ORA_NLS (Oracle 7.2 only) ORA_NLS32 (Oracle 7.3 only) ORA_NLS33 (Oracle 8.x only)	Set to the NLS data path (manually expand $ORACLE_HOME). $ORACLE_HOME/ocommon/nls/admin/data

The appropriate Oracle NLS data files (for example, “All Languages”) must be installed for the character conversion to work.

The opcagt start-up script does not take effect until the next reboot. To temporarily set the OpC agent environment without rebooting, log on as root, make sure the Oracle NLS environment is set for your current session using “env,” and restart the OpC agents with the following commands:

```
opcagt -stop
opcagt -kill
opcagt -start
```

A File Names

This appendix lists the DB-SPI file names, along with their descriptions, owners, groups, and permissions. These file names fall into the following logical groups:

- Management Server files
- Management Server/managed node files
- Managed node files

Management Server Files

Templates for opccfgupld (OVO Upload Facility)

All files have permissions set to 444, owner=root and Group=sys following installation on the managed node.

Table 49 Management Server Files: Core

DBSPI-Core uploads files in: `/var/opt/OV/share/tmp/OpC_appl/DBSPI/core_set/C/` `/var/opt/OV/share/tmp/OpC_appl/DBSPI/core_set/ja_JP.PCK/` (Solaris) `/var/opt/OV/share/tmp/OpC_appl/DBSPI/core_set/ja_JP.SJIS/` (HP-UX)	
`core_set.idx` `core_apps_set.idx`	OVO upload control file. This file defines what is being uploaded into OVO: message groups, monitor templates, logfile templates, applications and template groups.
`TEMPLATES/MONITOR/core_monitor.dat`	Definition file for OVO templates: collector/analyzer program, external monitor for each metric and file system monitor. Contains all details for monitoring all metrics.
`TEMPLATES/TEMPLGROUP/tmplgrp`	Definition file for OVO template group structure.
`TEMPLATES/INTERFACE/core_dbspi_msg.dat`	Definition file for DB-SPI specific error message.

`APPLICATIONS/core_dbspi_apps.dat`
`/`
`core_nt_dbspi_apps.dat`
	Definition file for DBSPI-Core applications in the OVO application bank.

Table 49 Management Server Files: Core

TEMPLATES/TEMPLGROUP/ core_templgroup.dat	Definition file for DBSPI-Core OVO template group structure.
MSGGROUPS/core_msggroups.dat	Definition file for the OVO message group DBSPI
TEMPLATES/EC/DBSPIDedup.eco /DBSPIDedup.ecs /DBSPIState.eco /DBSPIState.ecs	event Correlation system files to create a "state" system.

Table 50 Management Server Files: Oracle

Oracle upload files in: /var/opt/OV/share/tmp/OpC_appl/DBSPI/ora_set/C/ /var/opt/OV/share/tmp/OpC_appl/DBSPI/ora_set/ja_JP.PCK/ (Solaris) /var/opt/OV/share/tmp/OpC_appl/DBSPI/ora_set/ja_JP.SJIS/ (HP-UX)	
ora_set.idx	OVO upload control file for the DBSPI-Oracle group. This file defines what is being uploaded into OVO: message groups, monitor templates, logfile templates, applications and template groups.
TEMPLATES/MONITOR/ora_monitor.dat	Definition file for OVO templates: collector/ analyzer program and external monitor for each metric. Contains all details for monitoring all Oracle metrics.
APPLICATIONS/ora_apps.dat	Definition file for DBSPI-Oracle applications in the OVO application bank.
APPLICATIONS/ora_reports.dat	Definition file for DBSPI-Oracle reports in the OVO application bank.
APPLICATIONS/ora_nt_reports.dat	Definition file for DBSPI-Oracle reports in the OVO application bank for NT.
TEMPLATES/TEMPLGROUP/ ora_templgroup.dat	Definition file for DBSPI-Oracle OVO template group structure.
MSGGROUPS/ora_msggroups.dat	Definition file for the OVO message group DBSPI
TEMPLATES/LOGFILE/ora_logfile.dat	Definition file for Oracle logfile encapsulation.
TEMPLATES/LOGFILE/ora_logfileNT.dat	Definition file for Oracle logfile encapsulation for NT.
APPLICATIONS/core_nt_dbspi_apps.dat	Definition file for DBSPI-Core applications for NT in the OVO application bank.
TEMPLATES/SCHEDULE/ora_schedule.dat /ora_scheduleNT.dat	Definition file for Service Reporter metrics.

Table 51 Management Server Files: Informix

Informix upload files in: `/var/opt/OV/share/tmp/OpC_appl/DBSPI/inf_set/C/` `/var/opt/OV/share/tmp/OpC_appl/DBSPI/inf_set/ja_JP.PCK/` (Solaris) `/var/opt/OV/share/tmp/OpC_appl/DBSPI/inf_set/ja_JP.SJIS/` (HP-UX)	
`inf_set.idx`	OVO upload control file for the DBSPI-Informix group. This file defines what is being uploaded into OVO: message groups, monitor templates, logfile templates, applications and template groups.
`TEMPLATES/MONITOR/inf_monitor.dat`	Definition file for OVO templates: collector/analyzer program and external monitor for each metric. Contains all details for monitoring all Informix metrics.
`APPLICATIONS/inf_apps.dat`	Definition file for DBSPI-Informix applications in the OVO application bank.
`APPLICATIONS/inf_reports.dat`	Definition file for DBSPI-Informix reports in the OVO application bank.
`TEMPLATES/TEMPLGROUP/inf_templgroup.dat`	Definition file for DBSPI-Informix OVO template group structure.
`MSGGROUPS/inf_msggroups.dat`	Definition file for the OVO message group DBSPI
`TEMPLATES/LOGFILE/inf_logfile.dat`	Definition file for Informix logfile encapsulation.
`TEMPLATES/SCHEDULE/inf_schedule.dat`	Definition file for Service Reporter metrics and for scheduled metric (22 & 24).

Table 52 Management Server Files: Sybase

Sybase upload files in: `/var/opt/OV/share/tmp/OpC_appl/DBSPI/syb_set/C/` `/var/opt/OV/share/tmp/OpC_appl/DBSPI/syb_set/ja_JP.PCK/` (Solaris) `/var/opt/OV/share/tmp/OpC_appl/DBSPI/syb_set/ja_JP.SJIS/` (HP-UX)	
`syb_set.idx`	OVO upload control file for the DBSPI-Sybase group. This file defines what is being uploaded into OVO: message groups, monitor templates, logfile templates, applications and template groups.
`TEMPLATES/MONITOR/syb_monitor.dat`	Definition file for OVO templates: collector/analyzer program and external monitor for each metric. Contains all details for monitoring all Sybase metrics.
`APPLICATIONS/syb_apps.dat`	Definition file for DBSPI-Sybase applications in the OVO application bank.
`APPLICATIONS/syb_reports.dat`	Definition file for DBSPI-Sybase reports in the OVO application bank.
`TEMPLATES/TEMPLGROUP/` `syb_templgroup.dat`	Definition file for DBSPI-Sybase OVO template group structure.
`MSGGROUPS/syb_msggroups.dat`	Definition file for the OVO message group DBSPI
`TEMPLATES/LOGFILE/syb_logfile.dat`	Definition file for Sybase logfile encapsulation.
`TEMPLATES/SCHEDULE/syb_schedule.dat`	Definition file for Service Reporter metrics.

Table 53 Management Server Files: MS SQL

Sybase upload files in: /var/opt/OV/share/tmp/OpC_appl/DBSPI/mss_set/C/ /var/opt/OV/share/tmp/OpC_appl/DBSPI/mss_set/ja_JP.PCK/ (Solaris) /var/opt/OV/share/tmp/OpC_appl/DBSPI/mss_set/ja_JP.SJIS/ (HP-UX)	
mss_set.idx	OVO upload control file for the DBSPI-MSS6 & DBSPI-MSS7 group. This file defines what is being uploaded into OVO: message groups, monitor templates, logfile templates, applications and template groups.
TEMPLATES/MONITOR/mss_monitor.dat	Definition file for OVO templates: collector/analyzer program and external monitor for each metric. Contains all details for monitoring all MS SQL metrics.
APPLICATIONS/mss_apps.dat	Definition file for DBSPI-MSSQL applications in the OVO application bank.
APPLICATIONS/mss_reports6.dat /mss_reports7.dat	Definition file for DBSPI-MSS6 & MSS7 reports in the OVO application bank.
TEMPLATES/TEMPLGROUP/ mss6_templgroup.dat / mss6_templgroup.dat	Definition file for DBSPI-MSS6 and DBSPI-MSS7 OVO template group structure.
MSGGROUPS/mss_msggroups.dat	Definition file for the OVO message group DBSPI
TEMPLATES/LOGFILE/mss_logfile.dat	Definition file for MS SQL logfile encapsulation.
TEMPLATES/SCHEDULE/mss_schedule.dat	Definition file for Service Reporter metrics.

Miscellaneous Other Files

All files have permissions set to 444, owner=root and Group=sys following installation on the managed node.

Table 54 Miscellaneous Other Management Server Files

File	Description
`/etc/opt/OV/share/conf/OpC/mgmt_sv/reports/C/dbspi_lc.sql`	Report that lists all nodes that have DBSPI templates assigned (for license check).
`/opt/OV/DBSPI/bin/dbspixterm`	Executes xterm for DBSPI UDM Config from the correct location, depending on the platform (OVO HP-UX or OVO Sun).
`/opt/OV/DBSPI/bin/dbspiudmck`	Checks the syntax of the UDM Config file.
`/opt/OV/DBSPI/bin/dbspicfgck`	Checks the syntax of a Configure DB Connections input file.

`/opt/OV/DBSPI/bin/dbspiorapv`
`/opt/OV/DBSPI/bin/dbspimssqlpv`
`/opt/OV/DBSPI/bin/dbspiinfpv`
`/opt/OV/DBSPI/bin/dbspisybpv`
	Script to run PerfView as metric operator actions and bring up graphs for Oracle, MS SqL, Informix, and Sybase.

`/opt/OV/DBSPI/bin/dbspi.mo`	OVO Sun message catalog file for the management server.
`/opt/OV/DBSPI/bin/dbspi.cat`	OVO HP-UX message catalog file for the management server.
`/opt/OV/DBSPI/conf/udm.cfg`	Contains the UDM configuration. This file is copied to the monitor directory for HP-UX, Linux, Solaris, AIX and NT on the management server so that it is automatically distributed to the managed nodes.
`/opt/OV/DBSPI/bin/dbspircfg.sh`	Script to bring up the remote configuration program for NT.
`/opt/OV/DBSPI/bin/dbspiudmcfg.sh`	Script to bring up UDM Config.
/opt/OV/DBSPI/reporter/DBSPI_SR.exe	Installs DB-SPI Service Reporter package on the NT system.
/opt/OV/DBSPI/demo/demora.tar	Used to demo DB-SPI (using Oracle as example). Instructions for the demo are in the DB-SPI `readme.txt` file.
/opt/OV/DBSPI/bin/inf_alarm	Reserved for future use.
/opt/OV/DBSPI/bin/dbspi_infap	Reserved for future use.

Management Server/Managed Node Files

The following files are installed on the management server when **swinstall** is run and installed on managed nodes when monitors, commands and actions are distributed from the management server to the managed nodes.

Monitor Files

All files in the table that follows are located on the management server according to operating system in compress ".Z" format to improve distribution performance:

- **HP-UX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /hp/s700/hp-ux10/monitor/`
- **Solaris NCS:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /sun/sparc/solaris/RPC_NCS/monitor/`
- **NCS AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /ibm/rs6000/aix/RPC_NCS/monitor/`
- **DCE AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /ibm/rs6000/aix/RPC_DCE_TCP/monitor/`
- **BBC AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /ibm/rs6000/aix5/RPC_BBC/monitor/`
- **Solaris DCE:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /sun/sparc/solaris/RPC_DCE_TCP/monitor/`
- **HP-UX 11.0 with ITO 5:** `/var/opt/OV/share/databases/OpC/mgd_node/customer /hp/pa-risc/hp-ux11/monitor/`

Monitor files are installed in the OVO monitor area on managed nodes as follows:

- **HP-UX, Linux, and Solaris managed nodes:** `/var/opt/OV/bin/OpC/monitor/`
- **AIX managed nodes:** `/var/lpp/OV/OpC/monitor/` (DCE agent)
  `/var/opt/OV/bin/instrumentation` (HTTPS agent)

All files have permissions set to 744, owner=root and Group=sys following installation on the managed node.

Table 55 Management Server/Managed Node Monitor Files (UNIX)

File	Description
dbspi.cat	Message file used by DB-SPI.
dbspi.mo	DB-SPI catalog file (Solaris only).
dbspi022.sql	Command files used for metric 3222.
dbspi1070	Script used by metric 1070 to get the last checkpoint duration.
dbspi_alert	Script used by dbspicao to determine the file size of the Oracle alert log (metric 66).
dbspi_fsfile	Script used by dpspicao to retrieve file system information on Oracle files (metrics 56, 58, 62, 64, 65, report 66).
dbspi_fsfree	Returns number of free blocks in a file system (metrics 0016).
dbspi_infis	Script to check for Informix instance status.
dbspi_infpc	Script to check for oninit Informix server processes
dbspi_lscnt	Script used by dpspicao to count trace and core files (metric 63).
dbspi_lsfile	Script used by dbspicao to list trace and core files (reports 58, 63, 66).
dbspi_lsnr	Determines if Oracle listener is running.
dbspi_mwcoll	Script that pipes data from the /var/opt/OV/dbspi/dsi/*.dat files to the MeasureWare data files in the same directories using DSI.
dbspi_pcheck	Process check script used by dbspicao (metric 0002).
dbspi_sybsp	Stored procedures used by dbspicas. These are loaded when dbspicas is first run on the system.
dbspi_tail	Script used by dpspicao to list the last 250 lines of the Oracle alert log (report 66).
dbspiadm.exe	DB-SPI NT application launcher.
dbspicai.sh	DB-SPI Collector/Analyzer script for Informix. The script determines the version of Informix and then calls the appropriate dbspicaix program.
dbspicai7	DB-SPI Collector/Analyzer program for Informix 7.2 and 7.3. Called by dbspicai when Informix 7.2 or 7.3 is detected.
dbspicai9	DB-SPI Collector/Analyzer program for Informix 9.x. Called by dbspicai when Informix 9.x is detected.
dbspicam.bat	Collector/analyzer batch file for MS SQL Server.

Table 55 Management Server/Managed Node Monitor Files (UNIX)

File	Description
dbspicao	DB-SPI Collector/Analyzer script for Oracle. The script determines the version of Oracle and then calls the appropriate dbspicaoxx program.
dbspicao72	DB-SPI Collector/Analyzer program for Oracle 7.2.x.x. Called by dbspicao when Oracle 7.2 is detected.
dbspicao73	DB-SPI Collector/Analyzer program for Oracle 7.3.x.x. Called by dbspicao when Oracle 7.3 is detected.
dbspicao80	DB-SPI Collector/Analyzer program for Oracle 8.x.x.x. Called by dbspicao when Oracle 8.x is detected.
dbspicas	DB-SPI Collector/Analyzer program for Sybase (all versions)
dbspifsmon	Script used by dbspicao to list Oracle processes (metric 0002).
dbspimss.exe	Collector/analyzer program for MS SQL Server.
dbspimwc	Script to run dbspi_mwcoll (DB-SPI MeasureWare collector).
dbspimwc.exe	MeasureWare collector program.
dbspiudm.cfg	Local configuration file for User Defined Metrics.

Monitor files are installed in the following directory on Windows managed nodes:

usr\OV\bin\OpC\monitor\ - NT managed node

Table 56 Management Server/Managed Node Monitor Files (NT)

File	Description
dbspico7.exe	Collector/Analyzer program for Oracle 7.
dbspico8.exe	Collector/Analyzer program for Oracle 8.
dbspicoi.exe	Collector/Analyzer program for Oracle 8.x
dbspicao.bat	Collector/Analyzer batch file to run either dbspico7.exe or dbspico8.exe.
dbspi.cat.Z	DB-SPI catalog file.
dbspiudm.cfg	Local configuration file for User Defined Metrics.

Command Files

Files in the following tables are located on the OVO management server in directories according to platform in compress ".Z" format to improve performance of distribution to managed nodes:

- **HP-UX:**
  /var/opt/OV/share/databases/OpC/mgd_node/customer
  /hp/s700/hp-ux10/cmds/

- **NCS Solaris:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc/solaris/RPC_NCS/cmds/`
- **NCS AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix/RPC_NCS/cmds/`
- **DCE AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix/RPC_DCE_TCP/cmds/`
- **BBC AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix5/RPC_BBC/cmds/`
- **OVO for HP-UX and Solaris:** `/var/opt/OV/bin/OpC/cmds/`
- **Solaris DCE:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc/solaris/RPC_DCE_TCP/cmds/`
- **HP-UX 11.0 with ITO 5** `/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/pa-risc/hp-ux11/cmds/`

Command files are located on managed nodes as follows:

- HP-UX, Linux, and Solaris managed nodes: `/var/opt/OV/bin/OpC/cmds`
- AIX managed nodes: `/var/lpp/OV/OpC/cmds/` (DCE agent)
  `/var/opt/OV/bin/instrumentation` (HTTPS agent)

All files have permissions set to 755, owner=root and Group=sys following installation on the managed node.

Table 57 Command Files (UNIX)

File	Description
`dbspi_mw_int`	Script run when DBSPI MW Enable application is executed to enable or initialize MeasureWare integration on the managed node.
`dbspi_mwclup`	Script run when DBSPI MW Disable application is executed to disable MeasureWare integration on the managed node.
`dbspiadm`	DB-SPI application launcher.
`dbspiali`	DBSPI Application Launcher for Informix specific applications defined in the OVO Application Bank.
`dbspialo`	DBSPI Application Launcher for Oracle specific applications defined in the OVO Application Bank.
`dbspials`	DBSPI Application Launcher for Sybase specific applications defined in the OVO Application Bank.
`dbspicfg`	Configure DB Connections program for configuring managed node databases and filters.
`dbspicfg.sh`	Script to execute dbspicfg program. and perform additional checks on local.cfg (filter test, database connect test, etc.).

Table 57 Command Files (UNIX)

File	Description
dbspicol	File used to turn on and off collection. For example: dbspicol ON enables collection dbspicol OFF disables collection
dbspiidom.sp	Reserved for future use
dbspiidud.sp	Reserved for future use.
dbspiiidud.fm	Reserved for future use.
dbspiindb.fm	Reserved for future use.
dbspiindb.sp	Reserved for future use.
dbspiinsv.fm	Informix DSI format file for server metrics.
dbspiinsv.sp	Informix DSI specification file for server metrics.
dbspiisom.sp	Informix DSI specification file.
dbspiisud.fm	Reserved for future use.
dbspiisud.sp	Reserved for future use.
dbspilogtail	UNIX actions used to obtain the end of Informix and Sybase log files to be placed in the annotations for the logfile message.
dbspioosm.sp	Reserved for Oracle Service Reporter metrics.
dbspiora.fmt	DSI format file, used by dbspi_mwcoll to describe the DBSPI Oracle data file for each database instance found in /var/opt/OV/dbspi/dsi/oracle/ *.dat
dbspiora.sp	Oracle DSI specification file used by dbspi_mw_int when initializing MeasureWare support.
dbspioraud.fm	Oracle DSI format file for User Defined Metrics.
dbspioraud.sp	Oracle DSI specification file for User Defined Metrics.
dbspisbdb.fm	Reserved for future use.
dbspisbdb.sp	Reserved for future use.
dbspisbdud.fm	Reserved for future use.
dbspisbsv.fm	Sybase DSI format file for server metrics.
dbspisbsv.sp	Sybase DSI specification file for server metrics.
dbspisbud.sp	Reserved for future use.
dbspisdom.sp	Reserved for Service Reporter metrics
dbspiseinfsh	Inserts Informix collector/analyzer entry into CRON for scheduled metrics 22 & 24.
dbspisel	Utility program to select a database instance (used by dbspialo).

Table 57 Command Files (UNIX)

File	Description
dbspiseschcl	Clears scheduled DB-SPI cron jobs.
dbspissom.sp	Reserved for Sybase Service Reporter metrics.
dbspissud.fm	Reserved for future use.
dbspissud.sp	Reserved for future use.
dbspiudmisp	Script allowing user to edit and save stored procedure definition files for Informix. All Informix stored procedures updated with this script must include **dbspi** string in their names.
dbspiudmosp	Script allowing user to edit and save stored procedure definition files for Oracle.
dbspiudmssp	Script allowing user to edit and save stored procedure definition files for Sybase. All Sybase stored procedures updated with this script must include **sp_UDM-DBSPI** string in their names.
dbspiverify	Command line script to execute DBSPI Verify.
dbspixterm	Executes xterm for DBSPI applications from the correct location, depending on the platform (HP-UX, Linux, AIX, Solaris).
dbspi_mw_con	Script to move A.03.00 DSI logfiles and create new DSI logfiles.

All files in this table are located in the following directory on the management server and are in compress ".Z" format to improve performance of distribution to managed nodes:

```
/var/opt/OV/share/databases/OpC/mgd_node/customer/ms
/intel/nt/RPC_DCE_TCP/cmds/ - NT managed node
```

Table 58 Command Files (NT)

File	Description
dbspicfg.exe	DBSPI Configuration program.
dbspicol.bat	File used to turn on and off collection. For example: `dbspicol ON` enables collection `dbspicol OFF` disables collection
dbspirptxxxx	Miscellaneous report definition files for MS SQL Server report launcher.
dbspirlm.bat	MS SQL Server report launcher batch file.
dbspixxxSQL	Miscellaneous SQL files for MS SQL report launcher.

Table 58 Command Files (NT)

File	Description
`dbspialm.bat`	MS SQL Server application launcher.
`dbspimcu.bat`	MeasureWare clean-up batch file.
`dbspixxxx.sp4` `dbspixxxx.fm`	MeasureWare spec and format files for Oracle 4 and MS SQL Server.

Action Files

All files in the table following are located in directories on the management server according to operating system and are in compress "`.Z`" format to improve distribution performance:

- **HP-UX**:
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/s700/hp-ux10/actions/`
- **NCS Solaris**:
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc/solaris/RPC_NCS/actions/`
- **NCS AIX:**
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix/RPC_NCS/actions/`
- **DCE AIX**:
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix/RPC_DCE_TCP/actions/ -`
- **BBC AIX:** `/var/opt/OV/share/databases/OpC/mgd_node/customer/ibm/rs6000/aix5/RPC_BBC/actions/`
- **Solaris DCE:**
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/sun/sparc/solaris/RPC_DCE_TCP/actions/`
- **HP-UX 11.0 with ITO 5:**
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/hp/pa-risc/hp-ux11/actions/`

Action files are located on the managed nodes as follows:

- OVO on HP-UX and Solaris managed nodes: `/var/opt/OV/bin/OpC/actions/`
- OVO on AIX managed nodes: `/var/lpp/OV/OpC/actions/` (DCE agent)
  `/var/opt/OV/bin/instrumentation` (HTTPS agent)"

All files have permissions set to 744, owner=root and Group=sys following installation on the managed node.

Table 59 Action Files

File	Description
`dbspifsmona`	Gets a list of large files for report.
`dbspi_063 (UNIX)`	Deletes old trace files for metric 63.
`dbspirli`	Report launcher script for Informix
`dbspirsl`	Report launcher script for Sybase.

Table 60 Action Files (NT)

File	Description
`dbspitrc.exe (NT)`	Displays the last lines of the MS SQL Server trace file.
`dbspi_063.exe (NT)`	Deletes old trace files for metric 63.

Managed Node Files

Monitor Templates

Templates distributed from the management server to the managed nodes in OVO, the monitor templates are encrypted and stored in the following file:

HP-UX, Linux, Solaris, Tru64:/var/opt/OV/conf/OpC/monitor
AIX: `/var/lpp/OV/conf/OpC/monitor`
NT: `\usr\OV\conf\OpC\<system>monitor\`

Data, Log, and Configuration Files

The data, log, and configuration files, listed in the table below, are located in:

- HP-UX, Linux, Solaris, Tru64: `/var/opt/OV/dbspi`
- AIX: `/var/lpp/OV/dbspi` (DCE agent)
  `/var/opt/OV/dbspi` (HTTPS agent)
- NT: `\usr\OV\dbspi`

All UNIX files have permissions set to 644, owner=root and Group=sys following installation on the managed node.

Table 61 Data, Log, and Configuration Files

File (location shown above)	Description
`local.cfg`	Configuration file which contains the information configured using Configure DB Connections (ORACLE, HOME, CONNECT, SID, Filters, etc.). This is a binary file with the CONNECT string encrypted.

`history/<tag_name_ instance_name_ m00[2-digit metric number]>`
For example:
`dbspi/history/openview_m0016`
	instance_name = database name xxxx = metric number, for example, `sap_m0018`, `sap_0019`, etc. These files contain information for performing delta calculations.

`history/<tag_name_ instance_name_r00 [2-digit metric number]>`
For example:
`dbspi/history/openview_r0016`
	Report files for automatic actions.

`dbtab`	This file contains information on the currently configured databases in plain text (type, name and home). It is used by the MeasureWare initialization script.
defaults	This local file can be used to add to or modify deployed template or configuration settings. For example, you can use it to turn data collection on/off for specific database instances.

Oracle MeasureWare Data Files

All files have permissions set to 644, owner=root and Group=sys following installation on the managed node.

Table 62 Oracle MeasureWare Data Files

File	Description

`dsi/oracle/<dbname>.dat or`
`dsi/informix/<server>/<server>.dat`
`dsi/sybase/<server>/<server>.dat`
`dsi\mssql\<server>\<server>.dat`
	Data that is collected by the collector/analyzer for each metric. It contains # of metrics + 1 values. Metrics which are not implemented or do not support MeasureWare have values of -1.

`dsi/oracle/<dbname> or`
`dsi/informix/<server>/<server>`
`dsi/sybase/<server>/<server>`
	Data loaded from the *.dat file using DSI.

`dsi/oracle/<dbname>.log or`
`dsi/informix/<server>/<server>.log`
`dsi/sybase/<server>/<server>.log`
`dsi\mssql\server\<server>.log`
	Data loaded from the *.dat file using DSI.

`dsi/oracle/dbname.log.desc or`
`dsi/informix/<server>/<server>.log.desc`
`dsi/sybase/<server>/<server>.log.desc`
`dsi\mssql\<server>\<server>.log.desc`
	Describes the DSI data and log files.

`dsi/oracle/<dbname>.log.spec or`
`dsi/informix/<server>/<server>.spec`
`dsi/sybase/<server>/<server>.spec`
	Describes the DSI data and log files.

Additional subdirectories may exist as follows:
`/osm/` for Service Reporter data files;
`/udm/` for user-defined metrics data files.

Logging and Trace Files

All files have permissions set to 644, owner=root and Group=sys following installation on the managed node.

Table 63 Logging and Trace Files

File	Description
`log/mw_int_status`	Results of the DBSPI MW Enable application to determine if MW integration was successfully enabled.
`log/mw_status`	Error information if MW integration is failing. If the debug=0 is set to debug=1 in /var/opt/OV/bin/OpC/monitor/dbspi_mwcoll, then information is written to this file each time the MW Collector runs (every 5 minutes).
`log/trace`	When tracing is turned on (using the DB-SPI Admin application Trace On), tracing and debug information is written to this file. This file can get very large if tracing is left on indefinitely.
`log/dbspierror`	Error information when the DBSPI collector/analyzer encounters an error condition. Messages written to this file are also sent to the OVO message browser.

B Components

This appendix covers the DB-SPI program components for Oracle, Informix, and Sybase and contains the following topics:

- Template Compatibility Matrix
- Templates
- Message Groups
- Core Applications

Template Compatibility Matrix

The following matrices show the compatibility of the template groups for Oracle, Informix, and Sybase to facilitate template selection during installation.

Figure 47 Oracle Template Compatibility Matrix

(2) Combine with 0 or more compatible:																		
(1) Choose One of:	Oracle: Quick Start	Oracle: Quick Start NT	Oracle: Standard Metrics	Oracle: Favorites	Oracle: SQL Net	Oracle: Drill Down	Oracle: Reporter	Oracle: Reporter NT	Oracle: Logfiles	Oracle Logfile Template	NT Oracle Logfile Template	Oracle: Listener	Oracle: Listener NT	Oracle: UDM Templates	Oracle: Oracle 8 Specific	DBSPI-Core	Oracle: OEM Templates
Oracle: Quick Start	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Oracle: Quick Start Remote	■	■	■	■	■	■	■	■	■	(2)	(2)	■	■	■	■	■	■
Oracle: Quick Start NT	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Oracle: Standard Metrics	■	■	■	■	■	■	(1)	(1)	■	(2)	(2)	(3)	(3)	■	■	■	■
Oracle: Favorites	■	■	■	■	■	■	(1)	(1)	■	(2)	(2)	(3)	(3)	■	■	■	■
Oracle: SQL Net	■	■	■	■	■	■	(1)	(1)	■	(2)	(2)	■	■	■	■	■	■
(1) Choose one of Reporter or Reporter NT																	
(2) Choose one of Logfile or NT Logfile																	
(3) Choose one of Listener or Listener NT																	
	■	=	Not compatible														
	■	=	Compatible														

Figure 48 Informix Template Compatibility Matrix

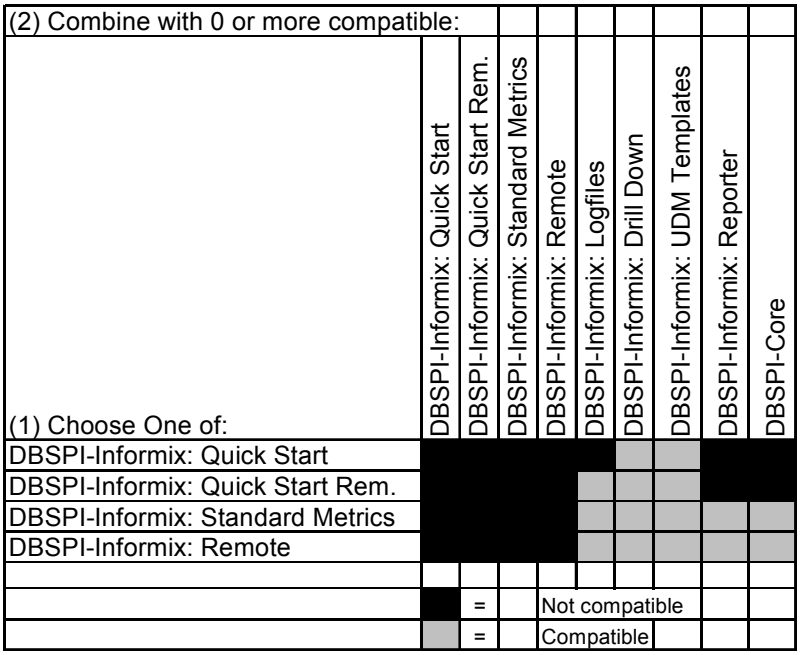


Figure 49 Sybase Template Compatibility Matrix

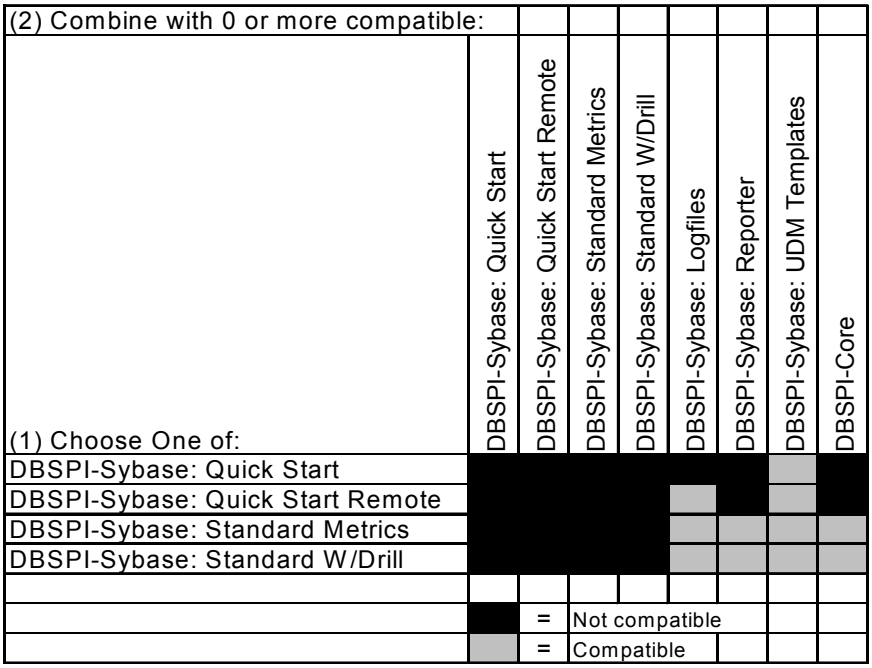


Figure 50 The Microsoft SQL Server 6.5 Template Compatibility Matrix

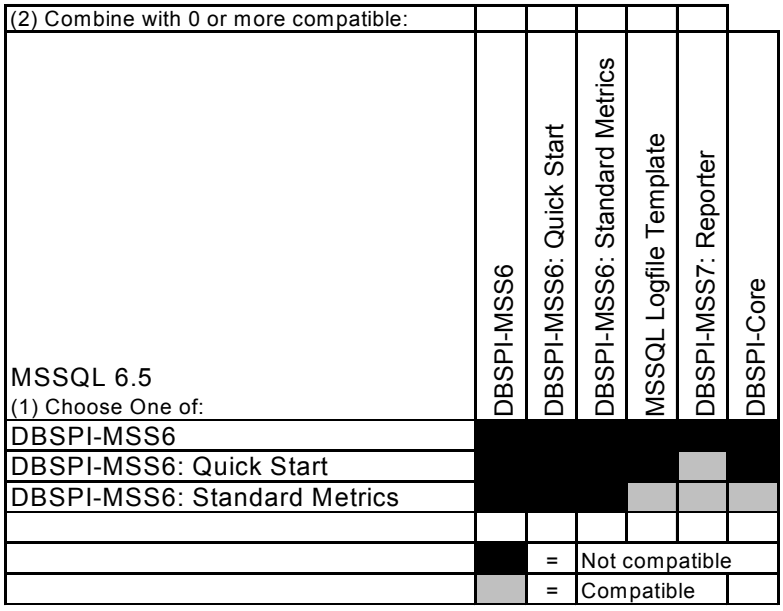


Figure 51 The Microsoft SQL Server 7 Template Compatibility Matrix

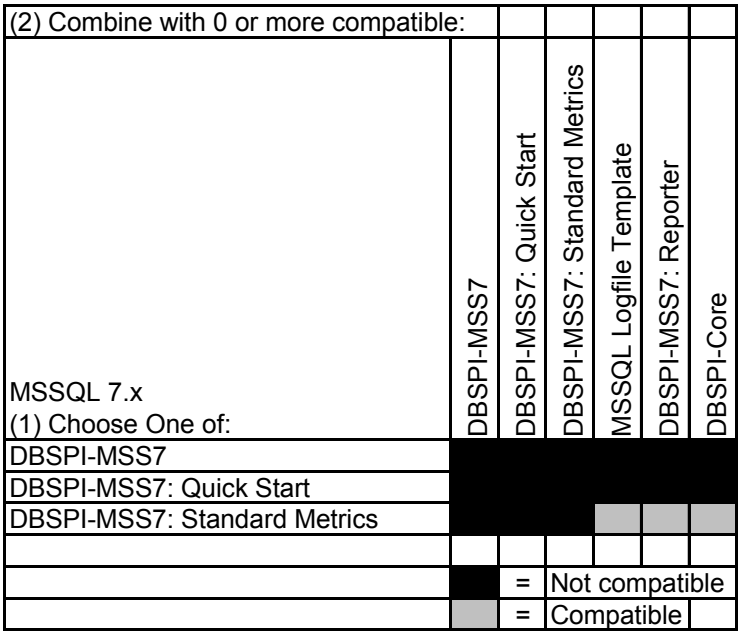


Figure 52 The Microsoft SQL Server 2005 Template Compatibility Matrix

(2) Combine with 0 or more compatible:						
MSSQL 2005 (1) Choose One of:	DBSPI-MSS2005	DBSPI-MSS7: Quick Start	DBSPI-MSS7: Standard Metrics	MSSQL Logfile Template	DBSPI-MSS7: Reporter	DBSPI-Core
DBSPI-MSS7						
DBSPI-MSS7: Quick Start						
DBSPI-MSS7: Standard Metrics						
		=	Not compatible			
		=	Compatible			

Templates

The Smart Plug-in for Databases template groups include those that apply to all databases, those that apply to the specific database you are using, and message groups, which classify the messages generated by other template groups. This section covers each group.

Common DB-SPI Templates

DB-SPI adds the following templates to the OVO Message Source Templates. These templates apply to all database applications:

— DBSPI-Core. This group is platform-independent and contains the following templates:

DBSPI-Messages. Contains the message file monitor for DB-SPI: DBSPI-Messages

DBSPI-MeasureWare. Contains the interface to OV Performance Agent (earlier called MeasureWare Agent): DBSPI-MeasureWare

DB-SPI Templates for Microsoft SQL Server

DB-SPI adds SQL Server templates to monitor MS SQL Server 6, 7, 2000, and 2005. The template groups are:

— DBSPI-MSS6 and DBSPI-MSS7*: Two groups contain the same template groups:

DBSPI-MSS6: Quick Start. Contains Core, Standard Metrics, and the Logfile and Reporter template subgroups.

DBSPI-MSS6: UDM Templates. Contains templates for configuring user defined metrics.

Can be used for MS SQL Sever 7, MS SQL 2000/2005.

Definitions (MSS6 & MSS7)

— DBSPI-MSS6 and DBSPI-MSS7: Standard Metrics. This group contains all standard MS SQL Server metrics, grouped as follows:

DBSPI-MSS6 & MSS7: By Collector (collector template & metrics)
DBSPI-MSS6 & MSS7: Metrics Only (complete list of metrics)
DBSPI-MSS6 & MSS7: Collectors Only (all MSSQL collector templates)

DB-SPI Templates for Oracle

DB-SPI adds these Oracle template groups to the OVO Message Source Templates:

— **DBSPI-Oracle.** Contains the following template groups (see next page for detailed information) and the listener template:

DBSPI-Oracle: Quick Start. Contains the subgroups DBSPI-Oracle Favorites, Core, Reporter; also contains the Alert Log monitor.

DBSPI-Oracle: Quick Start NT*. Contains the subgroups DBSPI-Oracle Favorites, Core, and Reporter; also contains the Alert Log monitor.*

DBSPI-Oracle: Standard Metrics. Contains all standard Oracle metrics.

DBSPI-Oracle: Quick Start Remote. Contains the templates for remote monitoring.

DBSPI-Oracle: Logfiles. Contains the Oracle alert logfile.

DBSPI-Oracle: UDM. Contains user-definable metrics.

DBSPI-Oracle: Oracle 8 & Higher. Contains monitor templates for metrics found only in Oracle versions 8 and higher.

DBSPI-Oracle: Listener. Contains DBSPI-Ora-Listener, DBSPI-Ora-Listener-Connect. (See Checking the Oracle Listener on page 119 for detailed instructions on how to make this work correctly; for UNIX only).

DBSPI-Oracle: Listener NT. Contains DBSPI-Ora-Listener-Connect. (See Checking the Oracle Listener on page 119 for detailed instructions on how to make this work correctly).

DBSPI-Oracle: OEM Templates. Contains OEM10G Message Interceptor Template.

Definitions

— **DBSPI-Oracle: Standard Metrics.** This group contains all standard Oracle metrics (does not contain drill-down or Oracle 8 metrics), including the following subgroups (details on each follow):

DBSPI-Oracle: By Collector
DBSPI-Oracle: Metrics Only
DBSPI-Oracle: Collectors Only

– **DBSPI-Oracle: By Collector.** This group contains standard Oracle metrics grouped by collection interval:

DBSPI-Oracle Collect 1d. This group contains the following collector and monitors:

DBSPI Ora-1d
DBSPI-0042, 0056, 0061

DBSPI-Oracle Collect 1h. This group contains the following collector and monitors:

DBSPI Ora-1h
DBSPI-0004, 0005, 0008, 0019, 0046, 0052, 0057, 0066, 0068, 0082

DBSPI-Oracle Collect 15min. This group contains the following collector and monitors:

DBSPI-Ora-15min
DBSPI-0016-18, 58, 62-65, 77-81

DBSPI-Oracle Collect 05min. This group contains the following collector and monitors:

DBSPI-Ora-05min
DBSPI-0001, 0002, 0006, 0007, 0009, 0011, 0020-0024, 0026-0028, 0031-0035, 0038-0040, 0043, 0045, 0048, 0050, 0054, 0059, 0060, 0067, 0069, 0075, 0083, 0085, 0087, 0089

– **DBSPI-Oracle: Metrics Only.** This group contains only Oracle metric monitor templates (no collectors):

DBSPI-0001, 0002, 0004-0009, 0011, 0016-0024, 0026-0028, 0031-0035, 0038-0040, 0042, 0043, 0045, 0046, 0048, 0050, 0052, 0054, 0056-0069, 0075, 0077-0083, 0085, 0087, 0089

- **DBSPI-Oracle: Collectors Only.** This group contains only Oracle collector templates (no metric monitors):

  DBSPI-Ora-05min
  DBSPI-Ora-15min
  DBSPI-Ora-1h
  DBSPI-Ora-1d

— **DBSPI-Oracle: Favorites.** This group contains the following collector templates for important ("favorite") metric monitors and the monitors:

DBSPI-Ora-05min-Favorites
DBSPI-Ora-15min-Favorites
DBSPI-Ora-1h-Favorites
DBSPI-Ora-1d-Favorites
DBSPI-0001, 0002, 0004-0009, 0016-0019, 0021-0024, 0026-0028, 0031-0035, 0042, 0043, 0056-0058, 0060-0062, 0064, 0065, 0067-0069, 0077-0080, 0082, 0083

— **DBSPI-Oracle: Listener**. For UNIX, contains DBSPI-Ora-Listener-UNIX, which checks to see whether the Oracle listener is running, and DBSPI-Ora-Listener-Connect, which connects to instance@instance. For NT, contains only DBSPI-Ora-Listener-Connect.

— **DBSPI-Oracle: Logfiles.** Contains the Oracle alert logfiles:

Oracle Alert Log Template
NT Oracle Alert Log Template

— **DBSPI-Oracle: Oracle 8 & Higher.** Contains monitor templates for metrics found only in Oracle versions 8 and higher:

DBSPI-Ora-15min-Oracle8
DBSPI-0070, 0071, 0074, 0076, 0090-0097

— **DBSPI-Oracle: Quick Start.** Contains the following:

DBSPI-Core
DBSPI-Oracle: Favorites
DBSPI-Oracle: Reporter
Oracle Alert Log Template

— **DBSPI-Oracle: Quick Start NT.** Contains the following.

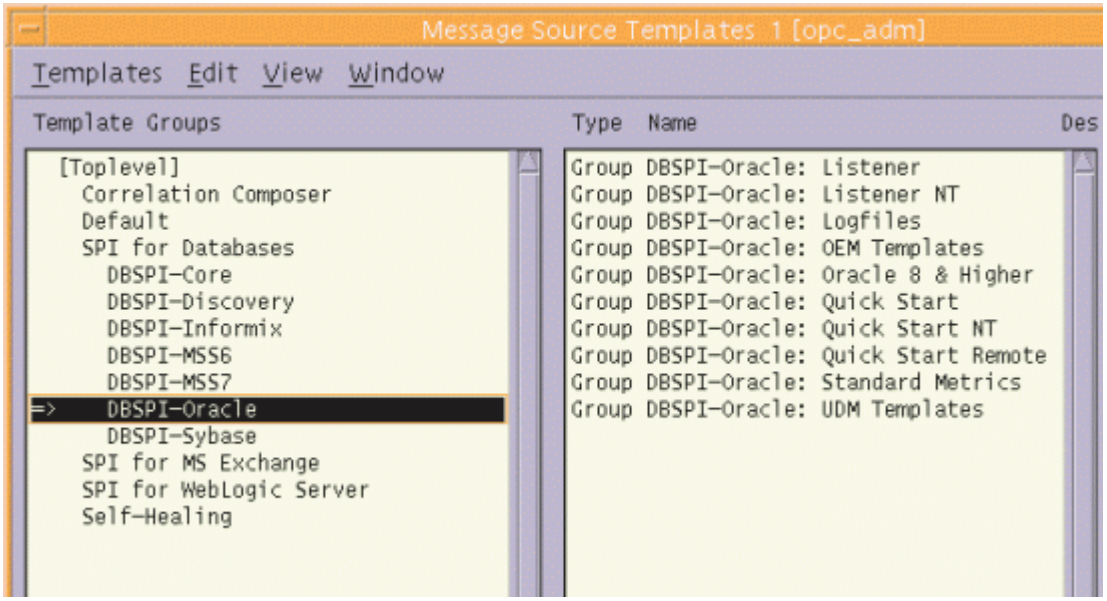
DBSPI-Core
DBSPI-Oracle: Favorites
DBSPI-Oracle: Reporter NT
NT Oracle Alert Log Template

— **DBSPI-Oracle: Quick Start Remote.** Contains the SQLNet collector templates and metrics that work with remote databases:

DBSPI-Ora-05min-SQLNet
DBSPI-Ora-15min-SQLNet
DBSPI-Ora-1h-SQLNet
DBSPI-Ora-1d-SQLNet
DBSPI-0001, 0004-0009, 0011, 0016-0024, 0026-0028, 0031-0035, 0038-0040, 0042-0043, 0045-0046, 0048, 0050, 0052, 0054, 0057, 0059-0061, 0067-0069, 0075, 0077-0083, 0085, 0087, 0089

— **DBSPI-Oracle: OEM Templates.** Contains OEM10G Message Interceptor Template. This template monitors the Oracle Enterprise Manager 10g environment.

Figure 53 DBSPI-Oracle Templates



DB-SPI Templates for Informix

DB-SPI adds the following Informix template groups to the OVO Message Source Templates:

— DBSPI-Informix. Contains the following template groups (see next page for definitions):

DBSPI-Informix: Quick Start. Contains DBSPI-Informix Standard Metrics, the MeasureWare collection template, the message file monitor, and the Informix logfile template.

DBSPI-Informix: Standard Metrics. Contains all standard Informix metrics and collector templates.

DBSPI-Informix: Logfiles. Contains the Informix logfile monitor.

DBSPI-Informix: Drill Down. Contains “drill-down” metrics and collector templates.

DBSPI-Informix: Remote. Contains metrics and collector templates that work with remote databases.

Definitions

— DBSPI-Informix: Quick Start. Contains DBSPI-Informix Standard Metrics, the MeasureWare interface, and the message file monitor:

DBSPI-Informix: Standard Metrics
DBSPI-Core
Informix Logfile Template

— DBSPI-Informix: Standard Metrics. This group contains all standard Informix metrics and collector templates, including the following subgroups (details on each follow):

DBSPI-Informix: By Collector
DBSPI-Informix: Metrics Only
DBSPI-Informix: Collectors Only

— DBSPI-Informix: Logfiles. Contains the Informix logfile monitor template.

— DBSPI-Informix: Drill Down. Contains “drill-down” metrics and collector templates. Drill-down metrics alarm once for each object violating a threshold:

DBSPI-Inf-05min-DrillDown
DBSPI-Inf-15min-DrillDown
DBSPI-1217, 1222, 1229, 1261

— DBSPI-Informix: Remote. Contains DBSPI-Informix metrics that work with remote databases:

DBSPI-Inf-05min-Remote
DBSPI-Inf-15min-Remote
DBSPI-Inf-1h-Remote
DBSPI-Inf-Daily-Remote
DBSPI-1003, 1005, 1007, 1008, 1015, 1017, 1020, 1022, 1024, 1031-1033, 1036, 1037, 1039, 1041, 1047, 1050, 1051-1053, 1059-1061, 1064-1066, 1069, 1071, 1073-1076

— DBSPI-Informix: By Collector. This group contains all Informix metrics grouped by collection interval:

DBSPI-Informix Collect 05min. This group contains the following collector and monitors:
DBSPI-Inf-05min
DBSPI-1001-1003, 1015-1017, 1020, 1032-1033, 1036-1037, 1039, 1047, 1061, 1064-1065, 1073-1076

DBSPI-Informix Collect 15min. This group contains the following collector and monitors:
DBSPI-Inf-15min
DBSPI-1005, 1007, 1022, 1024, 1029, 1031, 1059-1060, 1070-1071

DBSPI-Informix Collect Daily. This group contains the following collector:
DBSPI-Inf-Daily

DBSPI-Informix Collect 1h. This group contains the following collectors and monitors:
DBSPI-Inf-1h
DBSPI-1008, 1041, 1069

DBSPI-Informix: Collectors Only. This group contains only Informix collector templates (no metric monitors):
DBSPI-Inf-05min
DBSPI-Inf-15min
DBSPI-Inf-Daily
DBSPI-Inf-1h

DBSPI-Informix: Metrics Only. This group contains only Informix metric monitor templates (no collectors):
DBSPI-1001-1003, 1005, 1007, 1008, 1015, 1017, 1020, 1022, 1024, 1029, 1031-1033, 1036, 1037, 1039, 1041, 1047, 1059-1061, 1064-1066, 1069-1071, 1073-1076

DB-SPI Templates for Sybase

DB-SPI adds the following template groups to the OVO Message Source Templates:

— DBSPI-Sybase. Contains the following template groups:

DBSPI-Sybase: Quick Start. Contains DBSPI-Sybase Standard Metrics, the MeasureWare collection template, and the message file monitor.

DBSPI-Sybase: Standard Metrics. Contains all standard Sybase metrics and collector templates.

DBSPI-Sybase: Logfiles. Contains the Sybase logfile monitor.

DBSPI-Sybase: Standard W/Drill. Contains Sybase standard metrics with "drill-down" metrics and collector templates.

All DBSPI-Sybase metrics can be used to monitor remote databases.

Definitions

— DBSPI-Sybase: Quick Start. Contains DBSPI-Sybase Standard Metrics, the MeasureWare Agent interface, and the message file monitor:

DBSPI-Sybase: Standard Metrics W/Drill
DBSPI-Core
Sybase Logfile Template

— DBSPI-Sybase: Standard. This group contains all standard Sybase metrics and collector templates, including the following subgroups (details on each follow):

DBSPI-Sybase: By Collector —This group contains all Sybase metrics grouped by collection interval:

DBSPI-Sybase: Metrics Only
DBSPI-Sybase: Collectors Only
DBSPI-2016

DBSPI-Sybase: Collectors Only. This group contains only Sybase collector templates (no metric monitors):
DBSPI-Syb-05min
DBSPI-Syb-15min
DBSPI-Syb-1d
DBSPI-Syb-1h

DBSPI-Sybase: Metrics Only. This group contains only Sybase metric monitor templates (no collectors):
DBSPI-2001, 2003-2007, 2013, 2016, 2027, 2029, 2031-2033, 2036-2039, 2044-2048, 2050, 2051, 2053-2055, 2057, 2061-2076, 2080, 2081

DBSPI-Sybase: Collectors Only. This group contains only Sybase collector templates (no metric monitors):
DBSPI-Sybase Collect 05min
DBSPI-Sybase Collect 15min
DBSPI-Sybase Collect 1d
DBSPI-Sybase Collect 1h

DBSPI-Sybase: Standard Metrics. This group contains all standard Sybase metric monitor templates and collectors:
DBSPI-Sybase: By Collector
DBSPI-Sybase: Collectors Only
DBSPI-Sybase: Metrics Only

DBSPI-Sybase: Standard W/Drill. Contains standard Sybase metrics, “drill-down” metrics and collector templates. Drill-down metrics alarm once for each object violating a threshold:
DBSPI-Syb-05min-w/DrillDown
DBSPI-Syb-15min-w/DrillDown
DBSPI-Syb-1h-w/DrillDown
DBSPI-Syb-1d-w/DrillDown

DBSPI-Syb-05min-w/DrillDown
DBSPI-2001, 2003, 2007, 2013, 2027, 2029, 2032, 2033, 2035-2039, 2044-2048, 2050, 2051, 2053-2055, 2057, 2061-2076, 2080, 2081, 2203, 2213, 2235-2237

DBSPI-Syb-15min-w/DrillDown
DBSPI-2004, 2305, 2306, 2204

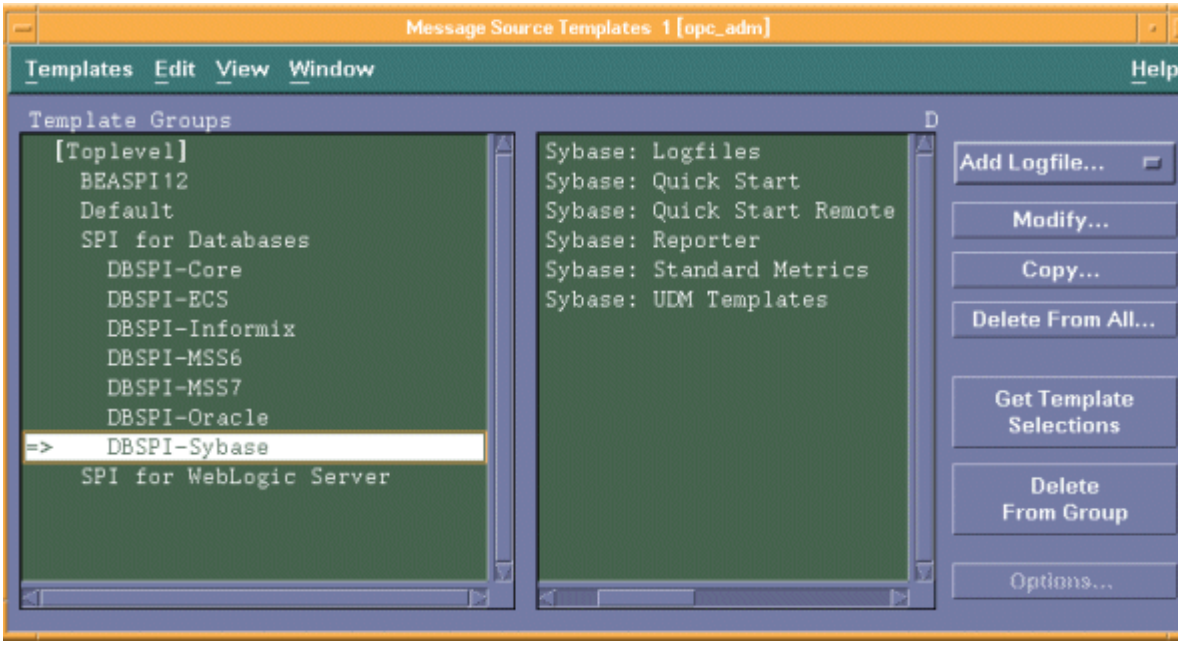
DBSPI Syb-1d-Drilldown
DBSPI-2031

DBSPI Syb-1h-Drilldown
DBSPI-2016, 2216

DBSPI-Sybase: Logfiles. Contains the Sybase logfile monitor.

See the *DB-SPI Sybase Reference* for details about the Sybase DB-SPI template groups.

Figure 54 DBSPI-Sybase Template Groups



DB-SPI Message Groups

DB-SPI adds the following platform-independent message group to the Message Group Bank:

— DBSPI. Messages generated by DB-SPI programs.They contain instruction text to help to diagnose/remedy problems.

Figure 55 Message Group Bank

DB-SPI Message Groups for Microsoft SQL Server

All messages generated by DB-SPI for SQL Server fall into one of the following message groups in the Message Group Bank:

MSS_Admin. Messages regarding SQL Server administration.

MSS_Conf. Messages regarding SQL Server configuration.

MSS_Fault. Messages regarding SQL Server error conditions.

MSS_Perf. Messages regarding SQL Server performance.

See the *HP OpenView Smart Plug-in for Microsoft SQL Server: Reference* to determine the message group for any SQL Server metric or logfile event.

DB-SPI Message Groups for Oracle

All messages generated by DB-SPI for Oracle fall into one of the following message groups in the Message Group Bank:

Ora_Admin. Messages regarding Oracle administration.

Ora_Conf. Messages regarding Oracle configuration.

Ora_Fault. Messages regarding Oracle error conditions.

Ora_Perf. Messages regarding Oracle performance.

See the *HP OpenView Smart Plug-in for Oracle: Reference* to determine the message group for any Oracle metric or logfile event.

DB-SPI Message Groups for Informix

All messages generated by DB-SPI for Informix fall into one of the following message groups in the Message Group Bank:

Inf_Admin. Messages regarding Informix administration.

Inf_Conf. Messages regarding Informix configuration.

Inf_Fault. Messages regarding Informix error conditions.

Inf_Perf. Messages regarding Informix performance.

See the *DB-SPI Informix Reference* to determine the message group for any Informix metric or logfile event.

DB-SPI Message Groups for Sybase

All messages generated by DB-SPI for Sybase fall into one of the following message groups in the Message Group Bank:

Syb_Admin. Messages regarding Sybase administration.

Syb_Conf. Messages regarding Sybase configuration.

Syb_Fault. Messages regarding Sybase error conditions.

Syb_Perf. Messages regarding Sybase performance.

See the *DB-SPI Sybase Reference* to determine the message group for any Sybase metric or logfile event.

Core (Administrative) Applications

DB-SPI adds the following applications that apply to all database applications to the Application Bank:

— DBSPI Admin. Contains configuration and troubleshooting applications requiring "root" permission, so it is recommended that the group be assigned to the OVO Administrator. This group contains the following applications:

Check Connections. Verifies connections to all databases configured on the managed node, useful in seeing that database location and database account information has been correctly configured. The metric collection monitor generates an error if it is unable to connect to a database.

Cleanup. Removes runtime files like trace, log, and persistent store files.

Configure DB Connections. Opens file with suggested syntax for configuring the managed node database. File entries define the database location, database account used by the DB-SPI to access the database; can also contain other settings for files/programs, such as the trace file and filtering parameters (also available for Windows managed nodes).

Configure Graphs. As needed, allows you to insert an entry in the configuration file for the location of the Web browser system on which the separately purchased OpenView Performance Manager is running.

Configure UDM. Opens the user-defined metrics configuration file with suggested syntax for configuring UDMs.

Create Oracle UDM. Supports the creation of stored procedures that enable user-definable metrics for Oracle.

Create Oracle User. (optional) Used to define Oracle user/password to access specific database instances on the managed node (must be implemented prior to DBSPI-Config, where database instances are defined).

Create SP/Informix UDM. Opens a file, where you can configure stored procedures for the creation of Informix user-defined metrics.

Create SP/Sybase UDM. Opens a file, where you can configure stored procedures for the creation of Oracle user-defined metrics.

Disable Graphs & Reports. Disables performance data collections including those used for graphing, UDM, and Reporter.

Display Error file. Displays the contents of the DB-SPI error file: `/var/opt/OV/dbspi/log/dbspierror`

Display Trace File. Displays contents of the DB-SPI trace file.

Drop Oracle User. Facilitates deleting Oracle user account.

Enable Graphs. Enables metric data collection for performance graphs.

Enable Reports. Enables collection of metrics specific to OpenView Reporter reporting (separately purchased product).

Enable UDM Graphs. Enables collection of UDM metrics specific to graphing (using the separately purchased OpenView Performance Manager).

Self-Healing Info. When run on the managed node, collects error message-related and OpenView Operations information that can be sent to HP support to troubleshoot SPI issues.

Start Monitoring. Re-enables performance metric data collection and alert notification. To turn collection back on at a command line:
UNIX: `/<OVO_commands_directory>/dbspicol -v ON`
`Windows: \<OVO_commands_directory>\dbspicol -v ON`.

Stop Monitoring. Temporarily disables performance metric data collection and alert notification. This application can be used to disable the metric collection during database maintenance, when using HP Serviceguard, or to minimize errors in the Message Browser if a database goes down unexpectedly. To turn collection off at a command line:
UNIX: `/<OVO_commands_directory>/dbspicol -v OFF`
`Windows: \<OVO_commands_directory>\dbspicol -v OFF`
(Paths for specific operating systems are shown at the beginning of Chapter 2.)

Trace Off. Turn DB-SPI tracing off.

Trace On. Turn DB-SPI tracing on.

Verify Deployment. Launches your Web browser (that you configured during the installation) so that you can select and view a graph, using (separately purchased) OpenView Performance Manager.

View Graphs. Shows deployed files, versions, number of templates, defaults file, and also performs a connection check.

— Admin Windows. Contains a subset of the DBSPI Admin applications for Windows managed nodes, including: Configure DB Connections, Check Connections, Verify Deployment, Display Error File, Start Monitoring, Stop Monitoring, Enable Graphs & Reports, Disable Graphs, Disable Reports, and Configure UDM.

Oracle Applications

Oracle applications can be run on UNIX nodes only, not on Windows nodes.

DBSPI Oracle requires that a user named `oracle` exist in group `dba`. If the user `Oracle` is not in group `dba`, you must modify **Customized Startup** to execute some Oracle applications. `Customized Startup` permits a different user login. Start/stop instance applications performs a `connect internal` requiring the executing user to be in group `dba`.

DB-SPI adds the following Oracle applications to the Application Bank:

— DBSPI Oracle. Contains Oracle-specific applications, which should be assigned to an OVO operator. Applications under DBSPI Oracle:

RAC Global Metrics (UNIX)
RAC Global Metrics (Windows). Allows choice of On/Off/Auto to configure metric collections for clustered databases.

SQL Plus*. Starts "sqlplus" program.

Svr Mgr (Text). Starts "svrmgrl" program (only available in Oracle >= 7.3)

Svr Mgr (X11). Starts "svrmgrm" program (only available in Oracle >= 7.3).

Start Instance. Starts a database instance.*

Start all Inst. Starts all configured database instances.*

Shutdown Inst. Shuts down a database instance.*

Shutdown all Inst. Shuts down all configured database instances.*

Shut. Inst. Immediate. Shuts down a database instance with "immediate" option.*

Shut. Inst. Abort. Shuts down a database instance with the "abort" option.*

SQL Net Status. Prints SQL Net status.

SQL Net Start. Starts SQL Net.

SQL Net Stop. Stops SQL Net.

Import. Calls the "imp" program for a database instance.

Export. Calls the "exp" program for a database instance.

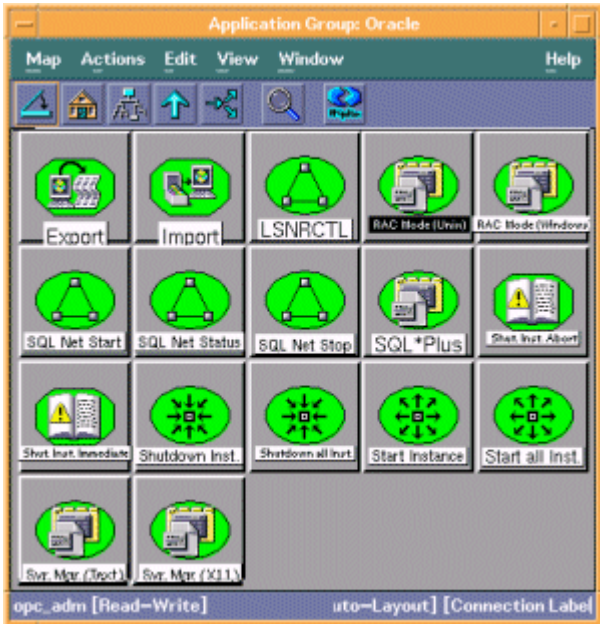
LSNRCTL. Calls the "lsnrctl" program for a database instance.7

*These applications require an internal connect to the database instance, so the executing account must have `dba` as its primary group (`Oracle`)

Oracle Reports. Reports on Oracle metrics 0004, 0005, 0006, 0007, 0008, 0009, 0011, 0016, 0017, 0018, 0029, 0031, 0037, 0038, 0042, 0045, 0047, 0057, 0058, 0063, 0066, 0067, 0068, 0069, 0077, 0078, 0079, 0080, 0081.

Oracle Reports Windows. Contains Oracle reports for Windows managed nodes. Reports on Oracle metrics 0004, 0005, 0006, 0007, 0008, 0009, 0011, 0016, 0017, 0018, 0029, 0031, 0037, 0038, 0042, 0045, 0047, 0057, 0058, 0066, 0067, 0068, 0069, 0077, 0078, 0079, 0080, 0081.

Figure 56 DBSPI-Oracle Applications



Informix Applications

DB-SPI adds the following Informix applications to the Application Bank:

— DBSPI Informix. Contains Informix-specific applications, which should be assigned to an OVO operator. Applications under DBSPI Informix

DB-Access. Runs sql statements, executes queries, creates and drops tables.

Find Error. Looks up error message text.

On-Check: Checks Root Reserved Pages, Checks System Catalog Tables, Checks Index Key Values

On-Monitor. Starts, stops database, sets database configuration parameters, and provides database views.

On-Perf. Graphs Informix performance data.

On-Stat. Provides status of Informix database.

Query Processor. Processes Xtree queries.

Server Status. Provides status information for a specified database server.

Server Status All. Provides status information for all Informix database servers found in dbtab.

Start DB Server. Starts a database server selected from list.

Start All Informix DB Servers. Starts all database servers found in dbtab.

Stop DB Server. Shuts down database server selected from list.

Stop All Informix DB Servers. Shuts down all database servers found in dbtab.

UNIX Login. Provides Unix login as Informix user.

DBSPI Informix Rep. Reports on Informix metrics DBSPI-1001, 1002, 1003, 1005, 1007, 1008, 1015, 1017, 1020, 1022, 1024, 1029, 1031, 1032, 1033, 1036, 1037, 1039, 1041, 1047, 1059, 1060, 1061, 1064, 1065, 1066, 1067, 1069, 1070, 1071, 1073, 1074, 1075, 1076.

Figure 57 DBSPI-Informix Report applications

Sybase Applications

DB-SPI adds the following Sybase applications to the OVO Application Bank:

— DBSPI Sybase. Contains Sybase-specific applications, which should be assigned to an OVO operator. Applications under DBSPI Sybase:

ISQL. Runs sql statements, executes stored procedures.

Start DB Server. Starts the selected data server.

Stop DB Server. Shuts down a selected data server.

Server Status. Returns status information for a selected data server.

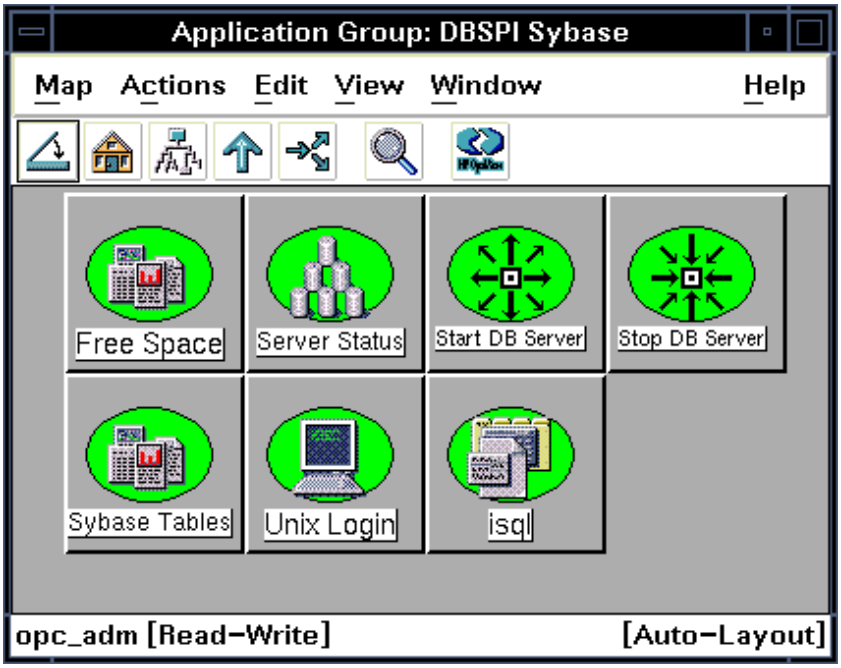
Server Status All. Returns status information for all data servers.

Sybase Tables. List of sybase tables on database server.

Free Space. Lists log size for each of the databases.

UNIX Login. Unix login as Sybase user.

Figure 58 Sybase Applications



Microsoft SQL Server Applications

DB-SPI adds the following MS SQL Server applications to the OVO Application Bank:

— DBSPI MS SQL. Contains MS SQL Server-specific applications, which should be assigned to an OVO operator. Applications under DBSPI MS SQL:

Active Jobs. Shows all jobs that are active.

All Jobs. Shows all active jobs as well as those idle and suspended, and completing.

Create MSSQL User. Allows user creation for the DB-SPI to connect to the MS-SQL Server for monitoring purposes.

NT Services. Shows NT services running.

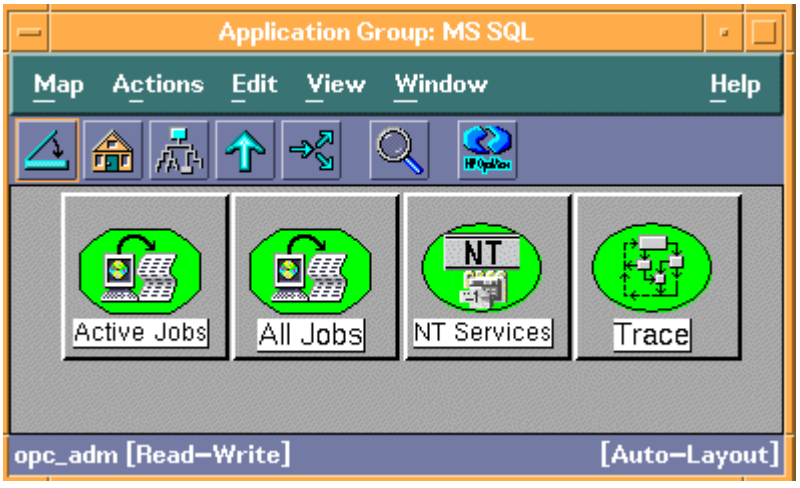
Start/Stop/List Instance. Starts, stops, or lists a specified MS SQL Server database instance.

Trace. Outputs the last 40 lines of the MS SQL Server logfile.

MS SQL jobs through the "`sp_help_job`" command has a job status according to number of the current_execution_status column:

Value	Meaning
1	Executing
2	Waiting For Thread
3	Between Retries
4	Idle
5	Suspended
7	Performing Completion Actions

Figure 59 MS SQL Applications



C Configuring Database Connections

Alternative Methods for Configuring Database Connections

The Database SPI application Configure DB Connections is the simplest way to open the configuration file. Other methods exist, though, if you are not at the OpenView Operations console, or you have a preference for working at the command line.

In some situations it may be necessary to set environment variables to enable a successful Database SPI connection to the database instance. For example, to address a situation where a Unix server did not have `ORACLE_BASE` below `ORACLE_HOME`, as expected, you could add an entry to the `defaults` file showing the `ORACLE_BASE` location:

```
SET ORACLE_BASE=/opt/ora9201
```

Prerequisite: Before you configure database connections by using any of the following methods, you must set the path to <*OVO_commands_directory*>. For more information about setting the path to <*OVO_commands_directory*>, see Running DB-SPI Applications from the Command Line on page 60.

Method 1: Configuring at the UNIX managed node:

Instead of using the Configure DB Connections application, log on to the managed node and execute:

```
dbspicfg.sh
```

When you open the configuration file for the first time, the file contains examples for keyword entries. When you save the file, the commented (#) lines are removed.

Method 2: Configuring at the Windows Managed Node

If you are at a Windows system and you would like to create the UNIX-compatible file from there:

1. At the Windows system, create the file with a text editor (see the section that follows for the proper syntax).
2. Save the file.
3. Open a Command Prompt window, and execute the commands:
   `dbspicfg.exe -i -u <input_file>`
4. Distribute the generated `local.cfg` and `dbtab` file to the UNIX managed node.

Method 3: Configuring Using FTP

If you are using systems on the other side of a firewall, xterm will likely not work. To work around this situation:

- For UNIX systems:

1. Open a telnet session to the system.
2. Configure the system using:
   `dbspicfg.sh`

- For Windows systems:

1. Open a telnet session to the system and configure the system using:
   `dbspicfg -e >file`
2. Edit the file:
   `dbspicfg -i <file`

Method 4: Configuring Multiple Nodes Simultaneously

An easy way to configure multiple network nodes follows.

DB-SPI stores the configuration information in a binary file at: `/var/opt/OV/dbspi/local.cfg`

1. Export the configuration file (stored in a binary file at `/var/opt/OV/dbspi/local.cfg`) to an ASCII file as follows:

   `dbspicfg -e >`*config_file*

   Where *config_file* is the name of the file that will contain the output.
2. Distribute this file to other nodes using FTP.

   (Be aware that this file contains passwords in plain text.)

   If the configuration on these nodes differs from the originating node, you can modify the *config_file* using an editor before importing it (see below). Or, you can import the file and then run **Configure DB Connections** against individual nodes to further customize the configuration.
3. Import the file from the *config_file* created above, as follows:

   `dbspicfg -i <config_file`

   Distributing the binary file `local.cfg` is not supported.

Configuration File Syntax

The following sections cover syntax for all entries for the configuration file as well as relevant information relating to the specific database type you are using. After discovery occurs on managed nodes, you will see that some configuration entries have already been provided.

Oracle

Oracle configuration file keywords: The table below describes Oracle keywords as used in the configuration file. See the example that follows for Oracle installations that include both non-ASM and ASM installations.

Table 64 Oracle UNIX node configurations

Oracle Keywords	Information (UNIX systems)	Double Quotes
HOME	Location where Oracle is installed	Yes
DATABASE	Name of the Oracle SID	No
CONNECT	DB user name and password in the form "*<userid>*/*<password>*" When configuring a UNIX ASM managed node, omit "*<userid>*" and "*<password>*" and include only the forward slash as in: `"/"` When configuring a multi-threaded server (MTS), include the connection alias: "*<userid>*/*<password>*@*<alias>*"	Yes
LISTENER	Used in conjunction with the DBSPI-Ora-Listener template to check on whether or not specific (named) listeners are running. `LISTENER` "*<name>*" If password protected: `LISTENER` "*<name>*" `CONNECT` "*<password>*"	Yes

Table 65 Oracle Windows node configurations (including ASM configurations)

Oracle Keywords	Information (Windows systems)	Double Quotes
HOME	Location where Oracle is installed	Yes
DATABASE	Name of the Oracle SID	No
CONNECT	DB user name and password in the form: "*<userid>*/*<password>*" When configuring a Windows ASM managed node, include "sys" and "*<sys_password>*" in the form: "*sys*/*<sys_password>*" When configuring a multi-threaded server (MTS), include the connection alias: *<userid>*/*<password>*@*<alias>*	Yes

Example entries for UNIX non-ASM and ASM configurations:

```
#Non-ASM Node Config (UNIX):
SYNTAX_VERSION 4
ORACLE
  HOME "/opt/oracle/product/10g"
    DATABASE openview CONNECT "hp_dbspi/password"

#ASM Node Config (UNIX):
ORA_ASM
  HOME "/opt/oracle/product/10g"
   DATABASE +ASM CONNECT "/"

#Non-ASM Node Config (Windows):
SYNTAX_VERSION 4
ORACLE
  HOME "c:\oracle\product\10.1.0\db_1"
    DATABASE openview CONNECT "hp_dbspi/password"

#ASM Node Config (Windows):
ORA_ASM
  HOME "c:\oracle\product\10.1.0\db_1"
   DATABASE +ASM CONNECT "sys/sys_pass"
```

For ASM installations on UNIX nodes that do not have `oracle` as the user name for the Oracle account, you must add the following entry the Database SPI `defaults` file: `UNIX_ORACLE_USER=`*<oracle_unix_account_name>*

This requirement does not apply to Oracle running on Windows.

Oracle Connect Clause: For the Oracle connect clause in the DB-SPI program you should enter or create an Oracle account that has permission to view the Oracle DBA and v$ performance view, create a session, alter a session (used by metric 57 report, DB-SPI tracing).

You can create an Oracle account for use by DB-SPI by using the DB-SPI application **Create Oracle User**, or you can create the account by executing a command. Either way, you open the same program that prompts you for the same entries.

Depending on the Oracle version you are using, some of the grants you create may generate reports of errors. This result is expected and is not a problem.

Lastly, though not recommended, you can use the Oracle `system` or `sys` account. The `sys` account has all permissions necessary to run all DB-SPI metrics and reports. If your database has locally managed tablespaces and you specify the `system` account, you must create the following grant as the `sys` user for metrics 16 and 216 to run without errors:

```
grant execute on sys.dbms_space_admin to system
```

You also need to create this grant if you see either of the following warning messages in the browser:

```
DBSPI40-22: Error [ORA-00904: invalid column name] occurred executing
dbms_space_admin - to fix as SYS do "grant execute on
sys.dbms_space_admin to <dbspiuser>".
```

OR

```
DBSPI40-22: Error [ORA-01031: insufficient privileges] occurred
executing dbms_space_admin - to fix as SYS do "grant execute on
sys.dbms_space_admin to <dbspiuser>".
```

To Create an Oracle Account with Specific Privileges

(For Oracle on UNIX systems use A or B. For Oracle on NT systems use C.)

A: To create an Oracle account for a UNIX system with the DB-SPI application:

1. Open the OVO Node Bank window and select the desired node.
2. Select Application Bank and in the Application Bank window double-click **DBSPI Admin** group.
3. Double-click the **Create Oracle User** application.

An xterm window running sqlplus appears. You are prompted for the information necessary to run the SQL script `dbspiocr.sql`, which creates a new Oracle account and grants the permissions necessary for DB-SPI to run all metrics and reports. For list of grants and privileges of Oracle user account, please refer to the file `dbspiocr.sql`.

For the default and temporary tablespaces, you can either accept the proposed tablespaces or use different ones. If you choose to modify tablespace sizes, the minimum *default* tablespace size is 96 KB, while the minimum *temporary* tablespace size is 48 KB.

B: To create an Oracle account for a UNIX system from the command line:

1. Log on as `Oracle` to the managed node.
2. Set your `ORACLE_HOME` and `ORACLE_SID` environment variables.

3 Type the following:
`cd` ***<OVO commands directory>***
`$ORACLE_HOME/bin/sqlplus /NOLOG @dbspiocr.sql`

You are prompted for the information necessary to run the SQL script **dbspiocr.sql**, which creates a new Oracle account and grants the permissions necessary for DB-SPI to run all metrics and reports.

C: To create an Oracle account for an NT system:

1 Start SQL*Plus on the NT OVO-managed node.

2 In the SQL*Plus window that appears enter the following:
`start \usr\OV\bin\OpC\cmds\dbspiocr.sql`

You are prompted for the information necessary to run the SQL script **dbspiocr.sql**, which creates a new Oracle account and grants the permissions necessary for DB-SPI to run all metrics and reports.

Oracle (UNIX) Connection Issue: DB-SPI connects to Oracle while logged on as user '`root`'. This sometimes causes connect problems in Oracle if certain operating system permissions are not set.

To verify that DB-SPI can connect to Oracle, you can do the following:

A. Log on as `root` (or system administrator for NT)

B. Set the following environment variables for your Oracle environment:

```
ORACLE_HOME
ORACLE_SID
```

C. Run **svrmgrl** (usually at $ORACLE_HOME/bin/svrmgrl) and connect to Oracle using the same user ID and password that you plan to use with DB-SPI by entering the command:

connect <*system*/*manager*>

Where *system*/*manager* is the user ID and password that you plan to configure for DB-SPI access.

If the connection fails and displays the following:

```
ORA-01034: ORACLE not available
ORA-07318: smsget: open error when opening sgadef.dbf file.
HP-UX Error: 13: Permission denied
```

OR

```
ORA-12546: TNS: permission denied
```

Ensure that `$ORACLE_HOME` and `$ORACLE_HOME/bin` have the permissions set to `drwxr-xr-x`

Ensure that the `$ORACLE_HOME/bin/oracle` file has the permissions set to `-rwsr-s--x`

If not set as above, enter this command:

```
chmod 6755 $ORACLE_HOME/bin/oracle
```

The following example shows how to include Oracle filter information in the Oracle DB-SPI Configuration File. See "Using Filters with DB-SPI on page 63" for filter specification syntax.

Figure 60 Filter examples for DB-SPI Configuration File

```
SYNTAX_VERSION 4

ORACLE

  HOME "/opt/oracle/product/7.3.3"
    DATABASE ora733 CONNECT "dbspi_user/password"
    FILTER 67 "segment_name <> 'RS1'"
    FILTER 4 "user_name NOT LIKE 'TEST%'"

  HOME "/opt/oracle/product/8.0.3/sw"
    DATABASE ora803 CONNECT "system/manager"
    FILTER 67 "segment_name <> 'RS1'"
    FILTER 6 "tablespace_name NOT LIKE 'SAP1001%' and
              tablespace_name NOT BETWEEN 'TS001' and
'TS010'"
```

MS SQL Server

No configuration necessary: If MS SQL Server was installed with default settings (using Windows authentication), it grants access to users that are part of the Administrator group. The HP OVO account (automatically created on the Windows node by OVO) is part of this group and therefore requires no special configuration to connect to the database.

The HP OVO account, under which the OVO agent is started, is part of the local Administrator group, meaning the collector/analyzer has immediate access to MS SQL server as a system administrator, and no special username/password setup is necessary. It is recommended to take advantage of this security approach if possible because NO configuration is needed and DB-SPI works as is.

Configuration necessary: If, however, MS SQL Server has been set up to manage its own authentication process for all connections, you must set up the HP OVO account access according to the MS SQL Sever access method used on the managed node, which can occur as follows:

MS SQL Server Node Configuration File	Access Method	HP OVO Account Requirement
No `local.cfg` is present, or no MSSQL section is defined in a configuration file.	Uses forced integrated login.	HP OVO account must be member of the group with SQL Server system administrator privileges.

MS SQL Server Node Configuration File	Access Method	HP OVO Account Requirement
A `local.cfg` is present and specifies: `MSSQL SERVER` *<server_name>*	Uses integrated login and connects to the specified server.	HP OVO account must be a member of the group with SQL Server system administrator privileges. For remote servers, the group must be a domain group with SQL Server system administrator privileges.
A `local.cfg` is present and specifies: `MSSQL SERVER` *<server_name>* `CONNECT` *<username/password>*	Uses standard SQL Server security.	No HP OVO account requirement. The specified username and password is used to log on and must have System Administrator privileges.

If it is necessary to configure the DB-SPI connection to MS SQL Server, use one of the two methods, using SQL Server keywords as listed in the table that follows.

MS SQL Server Keywords	Information	Double Quotes
SERVER	"" for local configuration. Name of server when more than one SQL Server is monitored.	Yes
CONNECT	DB user name and password in the form *<user>/<password>*	Yes

Example: MS SQL Server with Multiple Instances

```
MSSQL
SERVER "classact\instance1" CONNECT "<sa\mypassword>"
SERVER "classact\instance2" CONNECT "<sa\mypassword>"
SERVER "classact\instance3" CONNECT "<sa\mypassword>"
```

Method #1 (On the Windows Managed Node)

1. On the Windows managed node, use a text editor to create the configuration file syntax like the example below.

```
MSSQL
SERVER ""CONNECT "sa\mypassword"
MSSQL
SERVER "sapsvr1" CONNECT "sa\mypassword"
```

2. Save as a text file.
3. Input the configuration information into DBSPI by opening a Command Prompt window and entering:

```
c:\usr\OV\bin\OpC\intel\cmds\dbspicfg -i <file.txt>

To edit the file, you can export the existing file by entering:
c:\usr\OV\bin\OpC\intel\cmds\dbspicfg -e >file.txt
```

Method #2 (On the OVO Management Server)

1. On the OVO Management Server, double-click the NT-DBSPI Admin application group to open it.
2. Double-click the **DB-SPI Config** application.
3. Enter the following for local server configuration:

```
MSSQL
SERVER ""CONNECT "sa\mypass"
```

4. To connect to a remote server, add the server name as shown here:

```
MSSQL
SERVER "SAPSVR1"
  CONNECT "sa\mypassword"
```

To Create an MSSQL Account for Special Privileges

The following is a list of grants and privileges of MSSQL user account for MSSQL-DBSPI.

Database master:

GRANT select ON sysperfinfo

GRANT select ON sysprocesses

GRANT select ON syslocks

GRANT select ON sysconfigures

GRANT select ON sysperfinfo

GRANT select ON sysdatabases

GRANT select ON sysfiles

GRANT select ON sysindexes

GRANT select ON sysobjects

GRANT select ON sysdevices

GRANT execute ON sp_monitor

GRANT execute ON xp_sqlagent_enum_jobs

Database msdb:

GRANT select ON msdb..sysjobs

GRANT select ON msdb..sysjobhistory

GRANT select ON msdb..sysjobsteps

GRANT select ON msdb..sysjobschedules

Execute the stored procedure sp_grantdbaccess as follows on all the databases.

EXEC sp_grantdbaccess '<user>','<passwd>'

Following is the list of grants and privileges of MSSQL 2005 user account for MSSQL-DBSPI.

Database master:

GRANT create table

GRANT select ON sys.sysperfinfo

GRANT select ON sys.sysprocesses

GRANT select ON sys.dm_tran_locks

GRANT select ON sys.sysconfigures

GRANT select ON sys.databases

GRANT select ON sys.sysdatabases

GRANT select ON sys.sysfiles

GRANT select ON sys.sysindexes

GRANT select ON sys.sysobjects

GRANT select ON sys.sysdevices

GRANT execute ON sys.sp_monitor

GRANT execute ON sys.xp_sqlagent_enum_jobs

Database msdb:

GRANT select ON msdb..sysjobs

GRANT select ON msdb..sysjobhistory

GRANT select ON msdb..sysjobsteps

GRANT select ON msdb..sysjobschedules

GRANT select ON msdb..sysjobs_view

GRANT select ON msdb..MSdistributiondbs

Execute the stored procedure sp_grantdbaccess as follows on all the databases:

EXEC sp_grantdbaccess '<user>','<passwd>'

Grant the following to all the databases:

GRANT select on executionlog

Grant the following to all databases listed in the msdb..MSdistributiondbs:

GRANT select ON MSmerge_agents

GRANT select ON MSmerge_history

GRANT select ON MSmerge_sessions

GRANT select ON MSsnapshot_agents

GRANT select ON MSlogreader_agents

GRANT select ON MSdistribution_history

GRANT select ON MSsnapshot_history

GRANT select ON MSlogreader_history

GRANT select ON MSdistribution_agents

Informix

Informix Keywords: The table that follows describes keywords as used in the configuration file.

Informix Keywords	Information	Double Quotes
HOME	Location where Informix is installed	Yes
SERVER	Name of Informix server	Yes
ONCONFIG	Name of ONCONFIG file	Yes
CONNECT	UNIX user name and password in the form *<user>/<password>*	Yes
SQLHOSTS (optional)	Full path to SQLHOSTS file	Yes

Informix Connect Clause: The UNIX user specified in the Informix `Connect` clause must have certain capabilities and root must have certain capabilities. For most Informix installations, these capabilities are provided by the default Informix installation (i.e., no action is required). The capabilities required for the configured `Connect` user are:

```
Grant connect to sysmaster
Grant select to the sysmaster SMI tables
Execute permission on "onstat"
Execute permission on "dbaccess"
```

In addition, execute permission on "`onstat`" and "`dbaccess`" for the `root` user is required.

If the connection fails and you receive the message "Unable to load locale Categories," add two lines to the file `sbin/init.d/opcagt`:

```
CLIENT_LOCALE=en_US.819
DB_LOCALE=en_US.819
```

Try the connection again. These lines should eliminate the problem.

If the connection fails and displays the following error: `SQLCODE: -1829`
Verify that the file `citoxmsg.pam` exists in the directory `$INFORMIXDIR/msg`. If not, copy `$INFORMIXDIR/msg/itoxmsg.pam to $INFORMIXDIR/msg/citoxmsg.pam`.

Sybase

Sybase keywords: The table that follows describes keywords as used in the configuration file.

Sybase Keywords	Information	Double Quotes
HOME	Location where Sybase is installed	Yes
SERVER	Name of Sybase database server (from interfaces file)	Yes
REPSERVER	Name of Sybase replication server (from interfaces file)	Yes
CONNECT	DB user name and password in the form <*user*>/<*password*>	Yes
DATABASE (optional)	Name of specific database to monitor for metrics 2305, 2306, and 2405 only.	No
LOGFILE	Necessary if any Sybase logfiles are being monitored for proper logfile information reporting and autodiscovery of logfiles.	Yes
SYBASE_ASE	(optional) Name of the ASE directory. Example: `SBYASE_ASE: "ASE-12_5"`	Yes
SYBASE_OCS	(optional) Name of the OCS directory. Example: `SYBASE_OCS: "OCS-12_5"`	Yes

Sybase, 64-bit version requirement: For DB-SPI to work with a 64-bit version of Sybase, you must install the Open Client 32-bit libraries from Sybase. To do this, complete the following steps:

1. Obtain the Sybase product *Open Client*/*Server Product Version* <xx.y> for <*platform*>where <xx.y> is the version of Sybase (12.5, for example) and <*platform*> is the managed node platform, either HP-UX 11, Solaris, or AIX.
2. Use the *Open Client*/*Server Product* installation instructions to install the 32-bit libraries in the default library directory, usually:
   `$SYBASE/$SYBASE_OCS/lib`
   For example, the path might be `/home/sybase/OCS-12_5/lib` for Sybase 12.5

The 32-bit libraries do not conflict with already installed 64-bit libraries, since 64-bit libraries have "64" in them, while 32-bit libraries do not (that is, `libintl.a` is a 32-bit library; `libintl64.a` is the corresponding 64-bit library; `libintl.sl` is the 32-bit; `libintl64.sl` is the 64-bit, etc.).

Sybase Connect Clause: It is recommended that the Sybase `Connect` clause use the system administrator account, sa. DB-SPI requires some system administration functions to view performance information and to perform metric calculations. However, a non-`sa` account can be used, if the sa_role is granted to the DB-SPI account you create. The name and password you choose must be supplied in the Sybase `Connect` clause when you configure DB-SPI. Sample **isql** commands for creating a "dbspi" user with password "dbspiuser."

```
sp_addlogin dbspi, dbspiuser, tempdb
go
sp_role "grant", sa_role, dbspi
go
```

Sybase Connection Issue: If you are using OVO 5 and DBSPI-Sybase, verify that an entry exists in the `$SYBASE/locales/locales.dat` file corresponding to the OVO 5 localization value identified by the `START_LANG` variable in `/etc/rc.config.d/opcagt`.

For example: If the line `START_LANG=C.iso88591` is in the `opcagt` file, type the following entry in `$SYBASE/locales/locales.dat`

```
[hp ux]
; from hp:/usr/lib/nls/config, man hpnls
; from "Developing and Localizing International Software"
;       Hewlett-Packard Professional Books
;       ISBN 0-13-300674-3
locale = C.iso88591, us_english, roman8
locale = C, us_english, roman8
.
.
```

A localization variable mismatch can also be identified by the following errors in the message browser:

```
DBSPI 40-14: The call to function cs_ctx_alloc()(error=CS_FAIL) in
init_context failed.

DBSPI 40-14: The call to function init_context in ovam_sql_connect_db2
failed.
```

Configuration File

The file that follows illustrates how the configuration file appears when you open it, using the *Configure DB Connections* application (located in the Admin group). All commented lines (beginning with the pound [#] sign) are deleted after you save the file.

```
# Format of the file:
#
# SYNTAX_VERSION 4
# ORACLE
#   HOME "<ORACLE_HOME>"
#     DATABASE <name> CONNECT "<user/password>"
#       FILTER <number> "<SQL where qualifier>"
#     ...
#     DATABASE <name> CONNECT "<user/password>"
#       FILTER <number> "<SQL where qualifier>"
#     ...
#
# INFORMIX
#   HOME "<INFORMIX_HOME>"
#     SERVER "<name>"
#       ONCONFIG "<path/file>"
#       CONNECT  "<login/password>"
#       SQLHOSTS "<path/file>"
#       FILTER <number> "<SQL where qualifier>"
#     ...
# SYBASE
#   HOME "<SYBASE_HOME>"
#     SERVER "<name>"
#       CONNECT  "<user/password>"
#       LOGFILE "<path/file>"
#       FILTER <number> "<SQL where qualifier>"
#       DATABASE <name>
#
# NOTE - The LOGFILE entry is needed for logfile
# auto-discovery AND to be able to
# put critical logfile information into the OVO
# annotations (a tail is performed on the logfile
# and the information is placed in the Annotations
# for the browser entry).  So ALWAYS include a LOGFILE
# entry for Sybase to facilitate proper logfile
# encapsulation.
#

            #MSSQL

              SERVER "<name>" CONNECT "<sa\mypassword>"

              SERVER "<name>" CONNECT "<sa\mypassword>"
```

Examples

```
#(UNIX)
SYNTAX_VERSION 4
ORACLE

  HOME "/opt/oracle/product/8.1.7"
   DATABASE openview CONNECT "system/manager"
   LOGFILE "/opt/oracle/admin/openview/bdump/alert_openview.log"
```

```
SYBASE
  HOME "/opt/sybase"
    SERVER "sunbug" CONNECT "sa/sasasa"
       LOGFILE "/opt/sybase/ASE-12_0/install/sunbug.log"
    REPSERVER "REP_sunbug" CONNECT "REP_sunbug_RSSD_prim/sasasa"
       LOGFILE "/opt/sybase/REP-12_1/install/REP_sunbug.log"


INFORMIX
  HOME "/var/informix"
    SERVER "i914" ONCONFIG "onconfig.i914" CONNECT "informix/informix"
LOGFILE "/var/informix/online.log"

#(Windows)
ORACLE

  HOME "C:\oracle\ora90"
    DATABASE openview CONNECT "system/manager"
      LOGFILE "C:\oracle\admin\openview\bdump\openviewALRT.log"

MSSQL
  SERVER "ROS51340TST2"
  SERVER "ROS51340TST2\OVOPS"
```

Index

A

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V