Oracle® Enterprise Manager

Configuration Guide

Release 1.6.0

June 11, 1998

Part No. A63732-01

ORACLE®

Enabling the Information Age™

Oracle Enterprise Manager Configuration Guide, Release 1.6.0

Part No. A63732-01

Release 1.6.0

Contributing Authors: Eric Belden, Muralidharan Bhoopathy, Maureen Byrne, Erik DeMember, Jorge Fox, Jacqueline Gosselin, Todd Guay, Daniela Hansell, Marilyn Hollinger, John Kennedy, Dennis Lee, Priscilla Lee, Nahed Majzoub, Dimitris Nakos, Mark Osborn, Linda Pratt, Hanne Rasmussen, Geoffrey Rego, Bert Rich, Jonathan Riel, Marilyn Roncati, Gaylen Royal, Vipul Shah, and Jon Soule.

Contents

2 Console Configuration

Send Us Your Comments

Oracle Enterprise Manager Configuration Guide, Release 1.6.0

Part No. A63732-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail - infosmp@us.oracle.com
- FAX - (650) 506-7200. Attn: Oracle System Management Products
- Postal service:
  Oracle Corporation
  Oracle System Management Products Documentation Manager
  500 Oracle Parkway
  Redwood Shores, CA 94065
  U.S.A.

If you would like a reply, please give your name, address, and telephone number below.

If you have problems with the software, please contact your local Oracle World Wide Support Center.

Preface

This chapter describes the purpose and organization of this guide. The Preface contains the following information:

Purpose of this Guide

This guide explains how to configure Oracle Enterprise Manager. Oracle Enterprise Manager is Oracle's graphical administration and networking tool for distributed database environments.

For program updates, important notes on using Oracle Enterprise Manager, and information on the online documentation, see the *Oracle Enterprise Manager Readme*.

After you have completed the configuration procedures, see the Oracle Enterprise Manager online help or the *Oracle Enterprise Manager Administrator's Guide* for information on how to use Oracle Enterprise Manager.

Audience

This guide is written for DBAs and system administrators who wish to install Oracle Enterprise Manager.

How this Guide is Organized

This guide is comprised of the following chapters:

Chapter	Description
Chapter 1, "Agent Configuration"	This chapter describes the configuration of the Oracle Intelligent Agent.
Chapter 2, "Console Configuration"	This chapter describes methods for starting Oracle Enterprise Manager. This chapter also discusses connecting to a database.
Chapter 3, "Software Manager Configuration"	This chapter describes the installation procedure for Oracle Software Manager.
Chapter 4, "Pack Configuration"	This chapter describes the configuration of the applications in the Oracle Diagnostics Pack and the Oracle Tuning Pack.
Appendix A, "Configuration Files"	This appendix describes the files necessary for Oracle Enterprise Manager configuration.
Appendix B, "Directory Structure"	This appendix describes the directory structure of Oracle Enterprise Manager components and the Intelligent Agent.
Appendix C, "Hints and Tips"	This appendix describes helpful tips for the successful operation of Oracle Enterprise Manager.
Glossary	This glossary lists the definitions of some of the terms related to Oracle Enterprise Manager and its components.

Documentation Set

Because this guide only covers the configuration process, refer to the following publications for more information about the Oracle Enterprise Manager:

Oracle Enterprise Manager base documentation

- The *Oracle Enterprise Manager Readme* provides important notes regarding the online documentation, updates to the software, and other late-breaking information.
- The *Oracle Enterprise Manager Installation CD-ROM Insert* provides information about installing Oracle Enterprise Manager.
- The *Oracle Enterprise Manager Administrator's Guide* explains how to use Oracle Enterprise Manager, Oracle's systems management console, common services, and integrated platform tool.
- The *Oracle Enterprise Manager Concepts Guide* provides an overview of the Oracle Enterprise Manager.
- The *Oracle Enterprise Manager Application Developer's Guide* describes the programming external interfaces to the Oracle Enterprise Manager console.
- The *Oracle Enterprise Manager Messages Manual* describes the Oracle Enterprise Manager error messages and methods for diagnosing the messages.

Oracle Enterprise Manager Change Management Pack documentation

- The *Oracle Enterprise Manager Change Management Pack Readme* provides important notes regarding the Change Management Pack online documentation, updates to the software, and other late-breaking information.
- The *Oracle Enterprise Manager Getting Started with Oracle Change Management Pack* manual provides an overview of the concepts and features of the Oracle *Management Pack* applications.

Oracle Enterprise Manager Diagnostics Pack documentation

- The *Oracle Enterprise Manager Diagnostics Pack Readme* provides important notes regarding the Diagnostics Pack online documentation, updates to the software, and other late-breaking information.
- The *Oracle Enterprise Manager Getting Started with Oracle Performance Manager and Oracle Capacity Planner* manual provides an overview of the concepts and features of the Oracle Performance Manager and Oracle Capacity Planner applications.

- The *Oracle Enterprise Manager Oracle Trace User's Guide* explains how to use the Oracle Trace application to capture and use historical data to monitor Oracle databases.
- The *Oracle Enterprise Manager Oracle Trace Developer's Guide* explains how to instrument your application with Oracle Trace routines.
- The *Oracle Enterprise Manager Getting Started with Oracle TopSessions and Oracle Lock Manager* manual provides an overview of the concepts and features of the Oracle TopSessions and Oracle Lock Manager applications.

Oracle Enterprise Manager Tuning Pack documentation

- The *Oracle Enterprise Manager Tuning Pack Readme* provides important notes regarding the Tuning Pack online documentation, updates to the software, and other late-breaking information.
- The *Oracle Enterprise Manager Oracle Expert User's Guide* explains how to use Oracle Expert to optimize the performance of your database environment, during initial configuration, as well as during ongoing database operation.
- The *Oracle Enterprise Manager Getting Started with Oracle SQL*Analyze* manual provides an overview of the concepts and features of the Oracle SQL*Analyze application.
- The *Oracle Enterprise Manager Getting Started with Oracle Tablespace Manager* manual provides an overview of the concepts and features of the Oracle Tablespace Manager application.

The guides are available on the Oracle Enterprise Manager CD in HTML format for viewing with a web browser.

In addition to the Oracle Enterprise Manager documentation, extensive online help is provided for Oracle Enterprise Manager components.

Related Publications

The *Oracle Enterprise Manager Configuration Guide* refers to important information in the related publications. The related books referred to in this guide are listed below:

- For general information about the Oracle Server and how it works, see the *Oracle Server Concepts Guide*.
- For information about administering the Oracle Server, see the *Oracle Server Administrator's Guide*.
- For the procedures for migrating a previous version of Oracle, see the *Oracle Server Migration*.
- For information on Oracle's SQL commands and functions, see the *Oracle Server SQL Reference*.
- For information about utilities bundled with the Oracle Server, refer to the *Oracle Server SQL Reference.*
- For information about Oracle messages and codes, refer to *Oracle Server Messages.*
- For information about the Oracle networking system, see your network-specific documentation.
- For information about the Oracle Server on Windows NT, see the *Oracle Server for Windows NT Installation and User's Guide* and the system release bulletins, if available.
- For information on Oracle Parallel Server, please to the *Oracle Parallel Server Management Configuration Guide*, which provides essential information for preparing your Oracle8 Server for use with Oracle Parallel Server and Oracle Enterprise Manager. The Quick Start section lists all the server and console-related steps you need to perform to get started with the Oracle Parallel Server Management quickly.

  Note: This guide is directed at administrators setting up OPS on UNIX platforms only. For installation instructions, refer to your platform-specific installation guide.

  The *Oracle Parallel Server Management User's Guide* provides information on using the Oracle Enterprise Manager Console to monitor and control Oracle Parallel Servers.

Conventions Used in this Guide

This guide contains examples. This is an example of text that the user should type in a dialog box:

```
D:\WINDOWS.PRD
```

Note that the text of the examples appears in a different font than the text of this guide.

The examples in this guide use the following conventions:

- Examples of commands to be typed in by the user are in uppercase.
- Menu items and buttons referenced appear in initial uppercase.

1

Agent Configuration

The Oracle Intelligent Agents are processes running on remote nodes in the network. Oracle Enterprise Manager uses Intelligent Agents to run jobs, monitor events, and gather data on remote sites and to discover services on the node where it resides.

This Agent Configuration chapter discusses the following topics:

Please check the compatibility matrix in the Oracle Enterprise Manager Readme before installing or using Enterprise Manager. The `readme` is located in the `$ORACLE_HOME\SYSMAN\ADMIN` directory.

Installing the Intelligent Agent on Windows NT

Intelligent Agents are shipped with the database and installed on remote, managed machines. The Intelligent Agent must be installed in an ORACLE_HOME directory. The agent is installed as a service, called OracleAgent.

The Data Gatherer, which collects performance data used by the Oracle Capacity Planner and the new Java-based Oracle Performance Manager, is installed along with the Intelligent Agent. For information about configuring and upgrading the agent data gathering service, refer to *Configuring the Data Gathering Service* on page 1-18.

For information on installing the Intelligent Agent, please refer to the *Oracle Enterprise Manager Installation* (CD-ROM insert).

Configuring SNMP on Windows NT

Monitoring consoles use an SNMP Master Agent to communicate with the Intelligent Agent. The SNMP Master Agent and the Oracle Intelligent Agent must be configured correctly before the Oracle Intelligent Agent can communicate over SNMP to the Master Agent.

For the general procedures for configuring SNMP for Oracle databases and Oracle Enterprise Manager, refer to the *Oracle SNMP Support Reference Guide*.

For more comprehensive configuration information, see the installation or configuration guide specific to your platform since the SNMP configuration differs from platform to platform.

Configuring an Intelligent Agent on Windows NT

This section contains the following topics:

Agent Discovery Algorithm

At agent startup, a script is executed which reads configuration parameters from the Windows NT registry, `listener.ora` file, and the `tnsnames.ora` file (if it exists).

The agent discovers new services on the machine where it is installed and creates / rewrites / appends to its configuration files: `snmp_ro.ora`, `snmp_rw.ora`, and `services.ora`.

To determine what services are available on its machine (services that the agent will manage), the agent uses the following discovery algorithm:

1. The agent reads the Windows NT registry for the values of the services installed on that machine.
2. Based on values of the services, the agent reads the `listener.ora` files to determine the SIDs and ORACLE_HOME directories of the databases which are serviced by that listener.
3. The agent also reads the `listener.ora` files for the global names of those databases (`GLOBAL_DBNAME` parameters).
4. If `GLOBAL_DBNAME` parameters are not found in `listener.ora`, the agent searches for a `tnsnames.ora` file in a location set by the `TNS_ADMIN` environment variable.
5. If the `TNS_ADMIN` variable, which is either in environment or Registry under `ORACLE_HOME`, is not set, the agent looks for a `tnsnames.ora` file in the standard location, `$ORACLE_HOME\net80\ADMIN`.
6. If the `tnsnames.ora` file is not found, the database alias, `<SID>_<hostnames>`, is assigned to a database service. The service will be known to the agent by this alias, and it will be visible as such at the Oracle Enterprise Manager Console.

   If multiple aliases exist for the same instance in the `TNSNAMES.ORA`, the agent will use the one listed first. If you prefer to use a different alias, reorder the `TNSNAMES.ORA` entries and restart the Agent.

Note: If a database or any other new service is installed on the node where the agent resides, the agent must be restarted to add the new service to the agent configuration files.

Creating a Windows NT User Account for Running Jobs

In order for the agent to execute jobs on a managed node

- an NT user account must exist that has the advanced user right, "logon as batch job." The privilege can be assigned to an existing local or domain user, or a new NT user.
- the preferred credentials for the node must be set for that user in the Oracle Enterprise Manager console. Refer to *Setting Preferences* on page 2-12.
- the user must have read / write permissions to `ORACLE_HOME\NET80` directory as well as write permissions to the `TEMP` directory or the `ORACLE_HOME` directory.

Note: If you do not set up the "logon as batch job" privilege, you will receive the "Failed to authenticate user" message when you run jobs on the node.

Please follow one of the procedures listed below.

Creating a New NT User Account

To create a new Windows NT user account on the local NT machine (not the database) and grant the "log in as batch jobs" privilege to this user, perform the procedure below.

1. Select the User Manager from the Administrative Tools program group. See the Windows NT documentation for information on the tools.
2. Select New User from the User menu and check for the following:
   - The "User Must Change Password at the Next Logon" option box is not checked
   - "SYSTEM" or "system" is not used for the user name.
3. Under the Policies menu of the User Manager NT utility, select the User Rights option.
4. Check the "Show Advanced User Rights" box.
5. Select "Logon as a batch job" from the list of privileges.
6. Give the selected user this privilege.

Assigning Privileges to an Existing NT User Account

To assign privileges to an existing local user account, perform the following steps.

1. Choose the user on the User Manager panel and check for the following:
   - The "User Must Change Password at the Next Logon" option box is not checked
   - "SYSTEM" or "system" is not used for the user name.
2. Under the Policies menu of the User Manager NT utility, select the User Rights option.
3. Check the "Show Advanced User Rights" box.
4. Select "Logon as a batch job" from the list of privileges.
5. Add the advanced user right to this user.

Configuring a Domain User as Your Agent User

> **Note:** Domain users are not supported with 7.3.3 and earlier versions of the agent.

To configure a domain user as your agent user, perform the following steps.

1. Under the Policies menu of the User Manager NT utility, select the User Rights option.
2. Check the "Show Advanced User Rights" box.
3. Select "Logon as a batch job" from the list of privileges.
4. Click the Add button.
   a. Fill in the "List Names From" field: (choose your domain)
   b. Click Show Users button.
   c. In the listbox, choose the domain user.
   d. Click Add.
   e. Click OK.
5. In the User Rights Policy window, click OK.

Note: If you have both a local and a domain user with the same name, the local user takes precedence.

Controlling Operations of the NT Agent

This section contains information on controlling the agent through Windows NT and the DOS prompt. It also contains a section on troubleshooting the agent.

Note: Oracle Enterprise Manager and the agent use Net8 to communicate with the databases in question. Please verify that Net8 can connect to every SID in question before attempting to use the agent.

Starting the Intelligent Agent on Windows NT

To start the agent on Windows NT, perform the following steps:

1. Double-click the Services icon in the Control Panel folder.
2. Select the OracleAgent service.

   The Startup Type is set to Manual, which allows the agent to be started by a user. If you want the agent to start automatically whenever you start the system, set the Startup Type for Automatic.

   a. Click the Startup push-button. A Service Startup dialog box appears.

   b. Choose Automatic under the Startup Type.

   c. Click OK on the Service Startup dialog box.
3. Click the Start push-button to start the agent.

Stopping the Intelligent Agent on Windows NT

To stop the agent on Windows NT, perform the following steps:

1. Double-click the Services icon in the Control Panel folder.
2. Select the OracleAgent service.
3. Click the Stop push-button to stop the agent.

Verify Agent is Running

To verify that the agent is running, look for its status in the control panel services or type `netstart` at a command prompt. OracleAgent should appear in the list of services.

You may also view the NT Task Manager to see the `dbsnmp` process information.

Starting, Stopping, and Verifying the Status from the DOS prompt

To start or stop the Agent from the DOS command prompt, enter the appropriate command. From the DOS command prompt, you can also verify that the service is running.

If you want to...	Enter the following command
Start the agent from the DOS prompt	net start oracleagent
Stop the agent from the DOS prompt	net stop oracleagent
Verify status of the agent from the DOS prompt	`net start`

Troubleshooting the NT Agent If It Did Not Start Up

Make sure the agent service is up by checking the OracleAgent service in your control panel. If the agent did not start up, use any of the following hints listed below.

1. Check for messages written to the NT Event Viewer (under Administrative Tools) since this is where the NT agent writes any problems associated with startup.
2. Check if `snmp_ro.ora`, `snmp_rw.ora`, and `services.ora` are created by the agent on startup. `snmp_ro.ora` and `snmp_rw.ora` are in the `ORACLE_HOME\NET80\admin` directory, and `services.ora` is in the `ORACLE_HOME\NET80\agent` directory.

   Compare the services listed with the services which are available on the machine. Please refer to Appendix A, "Configuration Files" for valid sample files.

   If services are missing, check the following files for inconsistency or corruption:

   - `listener.ora`
   - `tnsnames.ora`
3. Check that you do not have a system path set to external drives.

   The agent is a service and runs by default as SYSTEM. It also needs DLLs from the `ORACLE_HOME/BIN` directory. If you need mapped drives in your path, you MUST NOT set them in the SYSTEM path.

   To set your own path:

   a. Move mapped drive paths out of SYSTEM path variables and into your own.

   b. Reboot to "unset" the systems path.
4. Check if you have TCP/IP installed. TCP/IP is a requirement.

5. If you still do not know why the agent did not start, trace the agent.
   a. Set the following variables in `snmp_rw.ora`:

      `nmi.trace_level=admin` (or 16 if you want maximum information)

      `nmi.trace_directory=`<any directory in which the Oracle user has write privileges>
   b. Restart the agent.
   c. Check the log files located in the `ORACLE_HOME/NET80/LOG` directory.

      `NMI.LOG` should show general agent problems.

      `NMICONFIG.LOG` should show problems with autodiscovery.
6. Ensure that the DNS Host entry is set to the node name in the listener.ora and tnsnames.ora files.
   a. Run the start button-> settings-> control panel-> network-> protocol-> TCP / IP properties.
   b. Check the DNS Host entry. For example, make sure that the entry is not set to the name of the previous engineer.
7. Turn on tracing for the daemon.
   a. Open `net80/admin/sqlnet.ora` and add the lines `daemon.trace_level=13` and `daemon.trace_directory=e:\orant\net80\trace`.
   b. Close the console to stop the daemon.
   c. Open the console to restart the daemon in trace mode.
   d. Submit a job and view the daemon.trc file for daemon and console problems.

Installing the Oracle Intelligent Agent on UNIX

Install the Oracle Intelligent Agent from the Oracle CD-ROM as per the *Oracle Enterprise Manager Installation Guide*. The Intelligent Agent is a separate component to select.

Running the root.sh Shell Script

After you have successfully installed the agent, the Oracle Installer prompts you to run `root.sh`.

`root.sh`, which is a shell script, updates/creates an `oratab` file. The `oratab` file is the file where the user will place references to all databases to be discovered by the agent and controlled by the Oracle Enterprise Manager. For each database created, the entry is of the form: `<SID>:<$ORACLE_HOME>:[Y/N]`

The agent is normally configured by `root.sh` as a `setuid` program. If `root.sh` was successful, the agent will have been installed as `setuid root` so that the agent can run jobs as the users whose name and password are given in the Preferred Credentials for that host.

To run a `setuid` program,

```
chown root dbsnmp
chmod 755 dbsnmp
```

If the agent is not a `setuid` program, all Enterprise Manager jobs are run with the permissions of the user who started the agent.

Note: The agent being set to `setuid` root does not have the same effect as having the `root` user start the agent. Having the `root` user start the agent may cause security problems. Consult your platform documentation for exact details on `setuid` programs.

The node credentials, the preferred credentials settings for the node, should be set to the user who starts the agent. It is the current username and password which appear when you click File-> Preferences and highlight the desired node name.

The user who submits node jobs to the UNIX node should ultimately be the owner of the ORACLE_HOME or the same user who started the Intelligent agent on UNIX. If the `root.sh` does not have the `setuid` set, then any job submitted to the agent will run with the privileges of the user indicated above. `root.sh` will allow the user to set the preferred credentials for that node to impose its privileges on the job.

Note: Please be aware that only one intelligent agent can be run on one UNIX machine although more than one Oracle Home can exist.

Note: If you have a 7.3.3 or below agent, installing an 8.0.x database is not recommended.

Verifying that root.sh Had Been Run Successfully

To verify that `root.sh` had been run successfully, check the file permissions on dbsnmp.

1. Enter `cd $ORACLE_HOME/bin`

   This changes the directory to the $ORACLE_HOME/bin directory where the agent executable resides.

2. Enter `ls -al dbsnmp`

   This lists all relevant details about `dbsnmp`.

The output of the `ls -al` command for dbsnmp should be in the form

```
-rwsr-xr-x   1 root     g651     1497980 Jun 12 21:04 dbsnmp
```

`root` is the owner. `dbsnmp` is the agent executable. In this example, the name of the group is `g651`. If `root` is the owner and `-rwsr-xr-x` are the permissions, then `root.sh` had been run successfully.

Running root.sh

Check the `oratab` file for entries for the database SIDs and Oracle Homes. Please refer to the *Agent Discovery Algorithm* on page 1-13 and your operating system-specific manual for the location of the `oratab` file.

Values of the Oracle Homes and SIDs should have been written by the `root.sh` shell script. If there are no entries in the `oratab` file, perform the following steps:

1. Switch user to root by typing

   su root

2. Change directory to the `$ORACLE_HOME/orainst` directory by typing

   cd $ORACLE_HOME/orainst

3. Run the `root.sh` shell script by typing

   ./root.sh

4. Answer the questions asked. For each database created, the entry is of the form: `<SID>:<$ORACLE_HOME>:[Y/N]`

You will then be automatically exited out of `root.sh`.

Configuring SNMP on UNIX

Monitoring consoles use an SNMP Master Agent to communicate with the Intelligent Agent. The SNMP Master Agent and the Oracle Intelligent Agent must be configured correctly before the Oracle Intelligent Agent can communicate over SNMP to the Master Agent.

For the general procedures for configuring SNMP for Oracle databases and Oracle Enterprise Manager, refer to the *Oracle SNMP Support Reference Guide*.

For more comprehensive configuration information, see the installation or configuration guide specific to your platform since the SNMP configuration differs from platform to platform.

Configuring an Intelligent Agent on UNIX

This section contains the following topics:

Agent Discovery Algorithm

At startup, the agent discovers new services on the machine where it is installed and creates its configuration files: `snmp_ro.ora`, `snmp_rw.ora`, and `services.ora`.

Note: Before attempting to start the agent, please test the Net8 connectivity by using the `tnsnames` aliases to connect to different SIDs. The agent depends on valid configurations of tnsnames.ora and listerner.ora. Refer to *Testing the Connectivity to Any SID* on page C-4.

To determine what services are available on its machine (services that the agent will manage), the agent uses the following discovery algorithm

1. The agent reads the `oratab` file for values of all the Oracle Homes and SIDs. Depending on the platform, the `oratab` file can be located in either of the following locations:
   - `/etc`
   - `/var/opt/oracle`
2. Based on the Oracle Homes values found in `oratab`, the agent reads the `listener.ora` files to determine which databases are serviced by that listener.
3. The agent also reads the `listener.ora` files for the global names of those databases (GLOBAL_DBNAME parameters).
4. If GLOBAL_DBNAME parameters are not found in the `listener.ora` file, the agent searches for `tnsnames.ora` files in a location set by the `TNS_ADMIN` environment variable. The standard location of the`tnsnames.ora` file is `$ORACLE_HOME/network/admin`. On Solaris the `tnsnames.ora` can be located in the `/var/opt/oracle` directory. On other UNIX systems, the `tnsnames.ora` can be located in the `/etc` directory. You can set a

$TNS_ADMIN environment variable if you do not want to use any of the default locations.

5. If the TNS_ADMIN variable is not set, the agent looks for tnsnames.ora files in the standard locations. For example: $ORACLE_HOME/NETWORK/admin.
6. If tnsnames.ora files are not found, the database aliases are created as <SID>_<HOSTNAME>.

   If multiple aliases exist for the same instance in the TNSNAMES.ORA, the agent will use the one listed first.If you prefer to use a different alias, reorder the TNSNAMES.ORA entries and restart the Agent.

Note: If a database or any other new service is installed on the node where the agent resides, the agent must be restarted to add the new service to the agent configuration file.

Setting the $ORATAB/ORATAB System Environment

It is recommended that you set the $ORATAB/ORATAB system environment variable before starting the agent.

The 733 and later agents on UNIX look at the ORATAB file to get a list of databases and their locations local to the server. The file contents look like this:

```
v732:/u01/oracle/product/7.3.2:n

v733:/u02/oracle/product/7.3.3:y
```

v732 and v733 are the names of the SIDs, followed by the ORACLE_HOME and then either a "y" or "n" for automatic startup.

To set the oratab environment variable ($ORATAB), type:

```
setenv ORATAB /etc/oratab
```

In the example above, ORATAB is the name you want to call the environment variable and /etc/oratab is the location and file to be used.

Controlling Operations of the UNIX Agent

On UNIX, Oracle Enterprise Manager uses the `lsnrctl` to start and stop the agent. The relevant `lsnrctl` commands to control the UNIX agent are listed in the table below.

If you want to...	Enter the command...
Start the agent on UNIX platforms	lsnrctl dbsnmp_start
Stop the agent on the UNIX platform	lsnrctl dbsnmp_stop
Verify status of the agent	lsnrctl dbsnmp_status

For additional information or restrictions for your platform, see the Intelligent Agent readme in `ORACLE_HOME/net80/agent/doc/readme.wri` or `ORACLE_HOME/network/agent/doc/readme.wri`.

Troubleshooting the UNIX Agent If It Did Not Start Up

Make sure agent listener is working. Enter the command:

```
lsnrctl dbsnmp_status
```

If the agent did not start up, use any of the following hints listed below.

1. Check the ORACLE_HOME/NETWORK/log/dbsnmp*.log file for errors on UNIX.
2. Check that the Oracle user has write permissions to ORACLE_HOME/AGENT/LOG as well as ORACLE_HOME/NETWORK/AGENT.
3. Check snmp_ro.ora, snmp_rw.ora, and services.ora for the entries created by the agent. snmp_ro.ora and snmp_rw.ora are in the ORACLE_HOME/NETWORK/ADMIN directory, and services.ora is in the ORACLE_HOME/NETWORK/AGENT directory.

   Compare the services listed with the services which are available on the machine. Please refer to Appendix A, "Configuration Files" for valid sample files.

   If services are missing, check the following files for inconsistency or corruption:

   - listener.ora
   - tnsnames.ora
   - oratab
4. If you still do not know why the agent did not start, trace the agent by setting the following variables in snmp_rw.ora:

   - nmi.trace_level=admin (or 16 if you want more information)
   - nmi.trace_directory=<any directory which the Oracle user can write to>
   - nmi.trace_file=agent

5. If you have upgraded the database software and one of your machines is having problems with the generated `snmp_ro.ora`, `snmp_rw.ora` or `services.ora` file, follow the instructions below:

   a. Run `catsnmp.sql` under the INTERNAL or SYS account (NOT the dbsnmp account). Normally the `catsnmp.sql` script is run from `catalog.sql` upon database creation but since this is an upgrade, you may not have run this script yet. If the necessary scripts have not been run, the dbsnmp account is not created.

   b. If you have more than one SID or older SIDs referenced in the `oratab`, run `catsnmp.sql` against each of them also.

   c. The `snmp_ro.ra` file is a read only file which means that all changes to the file will be overwritten each time the agent is started. You can make changes (if needed) to the `snmp_rw.ora` file.

   If you are trying to do backups, you must run `backupts.sql` with the `dbsnmp/dbsnmp` account.

Warning: Please do not modify the tcl scripts (job and events scripts written in Tool Command Language) which come with the agent. If you want to submit a job different than the ones that are predefined with the agent, use the TCL Job where you are allowed to pass in arbitrary scripts and have the agent execute them.

Configuring the Data Gathering Service

This section contains information on configuring the data gathering service for collecting Oracle database and host information.

Oracle Data Gatherer, which is part of the Oracle Enterprise Manager framework, collects performance data. This performance data is used by the Oracle Capacity Planner and the new Java-based Oracle Performance Manager. If you are not using either of these two products, you do not need to install and configure the agent data gathering service.

Requirements

The data gathering service (Oracle Data Gatherer) is intended to be located on the host where the monitored target (database or host) is physically located. In order to use the data gathering service to collect host-based performance statistics, Oracle Data Gatherer must be located on the monitored host.

However, in the case of collecting performance statistics for databases, Oracle Data Gatherer is not required to be installed and configured on the host where the database is located. Because it can collect database statistics remotely, Oracle Data Gatherer may be installed and configured on another node. Both the Oracle Capacity Planner and Oracle Performance Manager provide the ability to access performance statistics for a database through a data gathering service on an intermediate node.

It may be necessary or desirable to use this type of intermediate node data gatherer for the following scenarios:

- If the database to be monitored is a version before 8.0.4. Oracle Data Gatherer is introduced with 8.0.4 of the Oracle database and is not currently available in a configuration that can be installed with a pre-8.0.4 database.
- If an additional process footprint/overhead cannot be tolerated on the host where the monitored database is located. Of course the collection activity will still take place against the database, and that overhead will be present regardless of whether the data gathering service is located remotely or not.

Remember, however, that if Oracle Data Gatherer is not available on a host machine, then it is not currently possible to collect any host/operating-system performance statistics. This configuration limits data collection to database data on remote hosts.

Oracle Data Gatherer is installed along with the Intelligent Agent.

Upgrading the Data Gatherer

If you have a previous version of Oracle Data Gatherer installed and you have installed the new version of Oracle Data Gatherer into a different Oracle Home from the previous version, you should move the Capacity Planning configuration (called state files) and data files associated with the previous version of Oracle Data Gatherer to the Oracle Home where you have installed the new version of Oracle Data Gatherer.

The Oracle Data Gatherer state and data files are located in the `odg/reco` directory below the Oracle Home (`$ORACLE_HOME/odg/reco`) directory.

If you do not move these files, any binary data files created by Oracle Data Gatherer which have not yet been loaded into the Capacity Planner database will be lost, and the setup of what data is being collected for Capacity Planner will need to be entered into the Capacity Planner again following the installation of the new Oracle Data Gatherer.

It is not necessary to move any of these files if you have installed the new version of Oracle Data Gatherer into the same Oracle Home as the previous version or if you do not currently use the Oracle Capacity Planner.

Controlling Operations of the NT and UNIX Data Gathering Service

On UNIX and NT, Oracle Enterprise Manager uses the `vppcntl` command to manage the data gathering service. The `vppcntl` executable is located in `ORACLE_HOME/bin`.

Commands to control Oracle Data Gatherer are listed in the table below:

If you want to...	Enter the command...
Start Oracle Data Gatherer	`vppcntl -start`
Stop Oracle Data Gatherer	`vppcntl -stop`
Verify that Oracle Data Gatherer is running	`vppcntl -ping`
Identify the version of Oracle Data Gatherer	`vppcntl -version`

Additional Methods for Controlling Operations on Windows NT

This section contains information on controlling Oracle Data Gatherer through Windows NT and the DOS prompt.

To start Oracle Data Gatherer through the Control Panel on Windows NT, perform the following steps:

1. Double-click the Services icon in the Control Panel folder.
2. Select the OracleDataGatherer service.

   The Startup Type is set to Manual, which allows the data gathering service to be started by a user. If you want Oracle Data Gatherer to start automatically whenever you start the system, set the Startup Type for Automatic.

   a. Click the Startup push-button. A Service Startup dialog box appears.
   b. Choose Automatic under the Startup Type.
   c. Click OK on the Service Startup dialog box.
3. Click the Start push-button to start the data gathering service.

To start Oracle Data Gatherer from the DOS command prompt, enter the following command:

```
net start oracledatagatherer
```

To stop Oracle Data Gatherer through the Control Panel on Windows NT, perform the following steps:

1. Double-click the Services icon in the Control Panel folder.
2. Select the OracleDataGatherer service.
3. Click the Stop push-button to stop the data gathering service.

To stop Oracle Data Gatherer from the DOS command prompt, enter the following command:

```
net stop oracledatagatherer
```

Verify Data Gathering Service is Running

On Windows NT, use `vppcntl -ping` to verify if Oracle Data Gatherer is running. Then, look for its status in the control panel services. You may also view the NT Task Manager to see the vppdc process information. The data gathering service's message log is `ORACLE_HOME\odg\bin\alert_dg.log`.

On UNIX, use `vppcntl -ping` to verify if Oracle Data Gatherer is running. The data gathering service's message log is `$ORACLE_HOME/odg/bin/alert_dg.log`.

Using `vppcntl -ping` or checking the status of the OracleDataGatherer service in the control panel are different ways of checking if Oracle Data Gatherer is running. `vppcntl -ping` is the preferred way of checking the status of Oracle Data Gatherer.

Oracle Intelligent Agent and Oracle Names

The Intelligent Agent 8.0.4 does not use Oracle Names to discover services it manages. It uses GLOBAL_DBNAME parameters in `listener.ora` files to determine which databases that listener services. The Oracle Enterprise Manager console then uses the GLOBAL_DBNAME parameters to name the database in the Oracle Enterprise Manager Console.

The GLOBAL_DBNAME parameter typically describes the name of the database as it is registered with the Names Server, for example, the name and domain of the database as given in the database initialization parameter file. Values of the GLOBAL_DBNAME parameters must be unique.

If you are running Oracle Names on a machine managed by an Oracle Intelligent Agent, it is assumed that the databases have already been registered with a Names Server and their aliases are defined by the GLOBAL_DBNAME parameters in the `listener.ora` files.

Note: Please make sure that you are connected to this DBNAME and SID by testing the Net8 connections on that same server. Check the Net8 configuration for further information.

When running jobs or monitoring events in this environment, the Intelligent Agent does not resolve database aliases via Oracle Names.

For more information on how the Enterprise Manager Console works with Oracle Names, see Chapter 2, "Console Configuration".

Note: If you are planning to manage two or more Oracle databases on the same node, make sure the GLOBAL_DBNAME parameters in your `listener.ora` files are different for all databases.

Roles and Users Required by the Agent

The catsnmp.sql script is only installed when you install the database. When an Oracle database is installed, the catsnmp.sql script is automatically run by catalog.sql to create the necessary dbsnmp user account (the user dbsnmp with password dbsnmp) and the SNMPAGENT role for the Intelligent Agent (for 7.3.3 and later).

For security reasons, the customer may need to change the user / password for the Intelligent Agent's database logon. The default account is dbsnmp and the default password is dbsnmp. To change the user name and password to something other than dbsnmp / dbsnmp, you need to open, edit, and rerun catsnmp.sql for your own user and password before you edit the snmp_rw.ora.

The following parameters (connect lines) are needed if you have modified the account and / or password for the agent. You may add them to the snmp_rw.ora file in the ORACLE_HOME/NET80/ADMIN when you run the catsnmp.sql script:

```
SNMP.CONNECT.<svcname>.NAME = <USERNAME>
SNMP.CONNECT.<svcname>.PASSWORD = <password>
```

To determine whether the SNMPAGENT role exists in a database, enter the following SQL command:

```
SELECT * FROM dba_roles;
```

If the SNMPAGENT role does not appear, run the catsnmp.sql script on the database.

If you already have several versions of the database running, you must run the catsnmp.sql script on each of these database in order to have the correct setup for all the grants and views the agent needs to contact.

To run the script, you must log in as SYS or INTERNAL.

Note: The location of catsnmp.sql varies based on the database version you are running and the platform. For example, on NT for an Oracle 8.0.4 database, the script is located at ORACLE_HOME\rdbms73\admin. Please note that there is no harm in running the catsnmp.sql script more than once.

Creating Password Files for Oracle NT DB

Use the Password Utility ORAPWD80 to create password files. ORAPWD80 is automatically installed with the Oracle8 Utilities. Password files are located in the ORACLE_HOME\DATABASE directory and are named PWDSID.ORA, where SID identifies the Oracle8 database instance. Password files can be used for local or remote connections to an Oracle8database. The example below describes how to perform a local connection.

For example, to create a password file with ORAPWD80:

```
C:\> ORAPWD80 FILE=ORACLE_HOME\DATABASE\PWDSID.ORA PASSWORD=PASSWORD
```

2

Console Configuration

This chapter discusses getting started with Oracle Enterprise Manager and configuring the Console when it is running under Microsoft Windows NT or Windows 95.

Note: Please check the *Oracle Enterprise Manager Readme* for the compatibility matrix. The `readme` is located in the `ORACLE_HOME/SYSMAN/ADMIN` directory.

Setting Up the Repository User Account

Before you start the Enterprise Manager, you must set up the repository user account. A *repository* is a set of tables in an Oracle database which stores data and information required by the Oracle Enterprise Manager Console. The tables are owned by the account that creates them.

You need to have an Oracle user account with DBA (Database Administrator) privileges set up on the database where the repository will be created. Each DBA must have their own repository for the nodes they are responsible for. The repository account can be located in any Oracle database that is accessible on the network.

Please check the *Oracle Enterprise Manager Readme* for the compatibility matrix. The `readme` is located in the `$ORACLE_HOME\SYSMAN\ADMIN` directory.

Create a Separate Tablespace for the Repository

Oracle recommends that you create a separate tablespace for the Oracle Enterprise Manager repository.

1. Start Storage Manager.
2. Login to connect to an existing account on the database with system (DBA) privileges.
3. Select Create from the Tablespace menu. The Create Tablespace property sheet appears.
4. In the Name field, type `OEMREP` as the name of the new tablespace.
5. Click the Add button in the Datafiles section. The Create Datafile property sheet appears, allowing you to specify a new datafile which will belong to the new tablespace.
6. In the Name field, type in complete path and name of the datafile. For example: `c:\orant\database\oemrep01.dbf`.
7. In the Size section, enter the size of the new datafile where the size is calculated using the following formula:

   20M + DB_BLOCK_SIZE

For example, assuming a 2K database block size, the file size should be 20482K.

For the calculation, convert values to bytes; then, convert bytes to kbytes:

((20 * 1048576) + (2048)) / 1024 = 20482

1048576 is the number of bytes in a Megabyte

8. Click the OK button in the Create Datafile property sheet.
9. In the Extents Page, select Override Default Values and fill in the following information:

   Initial Size: 32K

   Next Size: 128K

   Increase Size By: 0

   Maximum Number:

   Select Unlimited if you are using Oracle 7.3 or later.

   Select Value if you are using a version prior to Oracle 7.3. Specify a value that is 75% of the maximum number of extents per segment as enforced by DB_BLOCK_SIZE.

   For example:

DB_BLOCK_SIZE	Enforced Maximum	Recommended Value
2K	121	90
4K	249	186
8K	505	378
16K	1017	762
32K	2041	1530

10. Click the Create button in the Create Tablespace property sheet.

Create a User

To set up a user account with DBA privileges, follow the procedure below:

1. Start Security Manager.
2. Login to connect to an existing account on the database with system (DBA) privileges.
3. Choose Create from the User menu. The Create User property sheet appears.
4. In the General Page, fill in the following information:

   In the Name field, enter a name for the new user.

   In the Password and Confirm Password fields, enter a password.

   Choose `OEMREP` as the default tablespace.

   Choose a tablespace that is dedicated to temporary segments as the user's temporary tablespace. In order to optimize sort performance, when running Oracle 7.3 or later, ensure this tablespace has been defined with the TEMPORARY keyword. For example, `temporary_data`.
5. In the Roles / Privileges Page, fill in the following information:

   Choose Role as the Privilege Type.

   Grant the DBA role to the user.
6. Click the Create button after specifying the requisite parameters.

   When you start Oracle Enterprise Manager, connect to the repository database with this user account. The repository will automatically be generated.

Note: Beginning with Oracle Enterprise Manager version 1.3.5, a new repository will be created for every new user login. One user can use one repository at one time.

Creating a Repository

The Repository Manager Wizard is started when you select Repository Manager in the Oracle Enterprise Manager program group.

> **Note:** You need to have an Oracle user account with DBA (Database Administrator) privileges in order to use the Oracle Repository Manager.

The Repository Manager allows you to choose from the following operations:

Operation	Description
Create / validate Enterprise Manager repositories	During the create / validate process, Repository Manager upgrades an existing repository or creates a new one if one does not already exist. If you drop a repository, the information contained in the repository is lost.
Drop Enterprise Manager repositories	If the database objects for the repository component exist, they will be deleted.
Save a list of the discovered nodes	You can choose to save a list of the discovered nodes in an existing repository to a text file. Specify the name, such as `discnode.txt`, and location of the file where you want to save the list of discovered node names. The default location is the `ORACLE_HOME\BIN` directory. The text file can be loaded into the Add Nodes page of the Discover New Services wizard.

You can choose the repository components that you want to create, validate, or drop. The Oracle Diagnostics Pack, Oracle Tuning Pack, and the Oracle Change Management Pack are dependent on the Oracle Enterprise Manager repository. Because of this dependency, a selection may be automatically checked and grayed out when another option is selected. For example, if you choose to create a Tuning Pack repository, the Oracle Enterprise Manager option is automatically selected because it is required.

When you choose Drop, Oracle Enterprise Manager is selected by default. The Oracle Diagnostics Pack, Oracle Tuning Pack, and the Oracle Change Management Pack are also selected because that repository cannot exist without the Oracle Enterprise Manager repository. If you want to drop only a specific pack's repository, remove the check on the Oracle Enterprise Manager repository; then, check the option for the pack you would like to remove.

The last page of the wizard allows you to enter the login information for the repository where the operation you have requested will be performed. The process takes a few minutes to complete.

All repository objects which are required by a particular tool will be created the first time the tool is started. Alternatively, the user can go into the repository and create all the repository objects at once.

If you have an existing repository from a previous release of Oracle Enterprise Manager, check for compatibility:

- If your Oracle Enterprise Manager repository is older than your Oracle Enterprise Manager Console, the existing repository is automatically upgraded when you log into the repository.
- If your Oracle Enterprise Manager Console is older than your Oracle Enterprise Manager repository, you must install a more recent compatible version of the Oracle Enterprise Manager. The console will detect this situation and inform you of the console version that is required.

If there is an unexpected failure during a repository upgrade, the repository will need to be recreated. To prevent having to recreate repositories at a future date, Oracle recommends that you save your repository. Saving the repository protects the user's manually entered information, such as events, jobs, and preferred credentials, as well as collected information for products such as Oracle Expert and Trace.

Repositories can be saved by doing a full cold backup (take the database off-line and copy datafiles to storage) or by exporting the user.

Starting Oracle Enterprise Manager

To start the Enterprise Manager Console, select Oracle Enterprise Manager from the Oracle Enterprise Manager folder on the Start Menu.

> **Note:** Only one instance of the Oracle Enterprise Manager Console can be run on a machine. However, you can run multiple instances of the DBA tools. Oracle Enterprise Manager is not supported to be run from a remote machine. It is only supported if run locally. Shared installs of the console are not supported.

Connecting to a Repository

Oracle Enterprise Manager requires unique usernames across all repositories whether or not the repositories are in different databases, because:

- The Intelligent Agents use the unique username when tracking jobs and events.
- The username is associated with the Console location when creating a return address for notifications.

The process for connecting to a repository is outlined below.

1. Enterprise Manager started.
2. The copyright window displays, then, the Login Information dialog box appears.
3. Through the Login Information dialog box, the user connects to the user account in the Oracle database where this Enterprise Manager repository is stored.
4. The Console looks up the repository associated with his username.

If the repository does not exist, it is automatically created and a dialog box informs you of the operation.

> **Note:** You should not log on to the Console repository multiple times with the same username. A warning displays if you attempt this. You should only ignore the warning if the previous Console session was aborted or a machine was disconnected. When a username is logged in multiple times, agent notifications are sent to the most recent login.

Discovering Network Services

The Oracle Enterprise Manager Console must discover network services, such as databases, listeners, and nodes, to populate the Navigator tree, in order to manage these services for the Job Scheduling and Event Management systems.

The Navigator's Service Discovery feature provides wizards for identifying network services and populating the Navigator tree.

To begin discovering services, your Oracle networking system must be properly installed. Methods of discovering services are listed below:

Method	Function	For Agent Version
Discover New Services Wizard	Discovers new nodes and the services on the nodes.	Intelligent Agent release 7.3.3 or higher using the TCP/IP protocol
Refresh Services Wizard	Refreshes and retries discovery on nodes	Intelligent Agent release 7.3.3 or higher using the TCP/IP protocol
Manually Define Services Wizard	Enables you to manually enter information about services on nodes	Does not have an Oracle Intelligent Agent or has a pre-7.3.3 Oracle Intelligent Agent
Discovery with `topo_ops.ora` file	Manual addition of Oracle Parallel Servers that cannot be added with the Navigator Service Discovery	Has a pre-7.3.3 Oracle Intelligent Agent

Note: The intelligent agent must be up at discovery time.

Important: If you have upgraded from a pre-8.0.4.1 Oracle Intelligent Agent, you must delete services already discovered and rediscovered them. The 8.0.4.1 and later agents support global database names.

To discover services:

1. Login to the Oracle Enterprise Manager Console.

   If this in a new install, the discovery wizard will appear.

   If this is not a new install, start the wizard. From the Main menu, select Navigator->Service Discovery->Discover New Services, and click the Next button.

2. In the Add Nodes Page, you can enter nodes one-by-one or load them from a text file.

   To enter nodes in the wizard, type the node name into the "New Node" field and click the Add button. Repeat for each node.

   If you have many nodes, you may want to construct two text files containing their names. The file should contain each node name on its own line. Name the text file *.text so that Oracle Enterprise Manager may find it.

3. Click the Next button.

4. Click the Finish button.

When requested through the Navigator Discovery menu option, the agent passes information from the `services.ora` file to the Enterprise Manager Console to populate the Navigator tree.

If you could not discover the database(s) in question, please refer to your platform-specific agent. For NT users, please refer to *Troubleshooting the NT Agent If It Did Not Start Up* on page 1-8. For UNIX users, please refer to *Troubleshooting the UNIX Agent If It Did Not Start Up* on page 1-16.

Note: Oracle Enterprise Manager no longer uses `topology.ora`. We recommend that you remove all of the services in the Navigator tree and delete the `topology.ora` file that was used by the previous release.

Since Oracle Enterprise Manager (beginning with version 1.4.0) no longer uses a `topology.ora` file for Pre-7.3.3 agents, the databases which do not have a 7.3.3 or higher agent running must be manually added to the navigator.

1. Log on to the Oracle Enterprise Manager Console.
2. From the main menu, select: Navigator->Service Discovery->Manually Define Services, and click the Next button.
3. Click on the box "This node contains a Pre-7.3.3 Oracle Intelligent Agent".
4. Enter the Node Name in the "Node Name" box and click the Add button.
5. Under Databases: Enter the database service descriptor defined in the `TNSNAMES.ORA` file on local box and on the server. The entries should match on both systems.
6. Click the Next button.
7. Click the Finish button.

Enterprise Manager Console and Oracle Names

Oracle Enterprise Manager Console and database applications can resolve service names via Oracle Names like any other Oracle Console application.

1. A client first looks in its own client-side cache. If it has contacted a Names Server within the past 24 hours, the address of a Names Server may be in the cache.
2. If a client has a `SQLNET.ORA` file, it looks for the NAMES.PREFERRED_SERVER parameter. If one exists, it queries the first Names Server listed.
3. A client looks for a Names Server with a well-known address. The well-known Names Server address on a TCP/IP network is a host aliased to oranamesrvr(), oranamesrvr1,...oranamesrvr4, using port 1575.

For more information on how to configure a client with Oracle Names, see the *Oracle Names Administrator's Guide*.

Note: Before attempting to discover these services or to run jobs, please test the Net8 connectivity. Please refer to *Testing the Connectivity to Any SID* on page C-4.

Starting Enterprise Manager DBA Applications

You can start an Enterprise Manager database administration application using one of the following methods:

If you are using...	Then...
Console	You can choose from the following three methods: ■ Select the database you want to administer in the Navigator tree or in the Map window, then choose the application in the Console Tools menu or in the launch palette. ■ Select the database you want to administer in the Navigator tree, then choose the application from the Related Tools menu of the context-sensitive menu. ■ Choose the application in the Console Tools menu or in the launch palette, then enter connect information in the Login Information dialog box.
Independent Application	**1.** Start the Enterprise Manager Console from the Start menu. **2.** Double-click on the application icon in the Enterprise Manager program group. **3.** Enter the connect information in the Login Information dialog box.
Administration Toolbar	Click on the tool's icon in the toolbar. The tool connects to the default database for the toolbar, if one has been specified. For information on connecting to other databases, see the online help for the Administration Toolbar.

Note: You can start the database administration applications without running the console. The only exceptions are Backup Manager and Data Manager which use the job system.

When a tool executes, it attempts to connect to an Oracle database. For information on connecting to an Oracle database, see the online help.

Before starting an application, when you select a database in the Console, you are connected to the database according to the preferred credentials that have been set up for the database or the credentials you used to log on to the Console. If connection to the database fails for any reason, the Login Information dialog box displays.

Setting Preferences

In order for the agent to execute jobs on a managed node, you must specify valid credentials in the Console.

Note: Jobs will be submitted using the repository owner login account unless the user sets the preferred credentials for the specific service (database, node, listener).

To set the preferred credentials, perform the following steps:

1. In the Oracle Enterprise Manager console, select File / Preferences for the services on the machine where the Intelligent Agent is installed
2. Specify the user account that you created as the user preference for the node where the agent is installed by highlighting that node name and entering the account name and password.

For NT users, you must set the preferred credentials for the node (where the NT Agent resides) to be the same as the user that is set up to "Logon as a batch job." The Agent authenticates the user name and password for all owners of jobs that it runs. If you have not set the preferences correctly, the "Failed to authenticate user" error occurs when you submit a job.

For UNIX users, you must specify a valid UNIX operating system user for the node (not db user for database). The user must have operating system privileges to run programs like sqlplus, import / export on the UNIX server. The privilege can be assigned to an existing local or domain user. The "Failed to authenticate user" error will occur if the preferred credentials do not match the account which started the agent.

Configuring a Remote Database for Backup or SYSDBA Administration

Before running the backup subsystem or submitting an administration task, such as database shutdown or startup, through the Oracle Enterprise Manager Job system, you need to set up your database for remote operation.

The following procedure outlines the operations you need to perform on both the server and client. The examples used in the following procedure are for a UNIX environment. Windows NT directories, system variables, and conventions will differ.

Additional Information: Refer to the operating system-specific documentation for your Oracle database for the exact name of the password file creation utility for your system.

Server Side (UNIX):

1. Login as ORACLE and set your environment variables.

   ```
   setenv ORACLE_HOME your_oracle_home
   setenv ORACLE_SID your_oracle_sid
   ```

   Note: This example assumes that you have set up your database as ORACLE.

2. Change to the dbs directory in your Oracle home.

   ```
   cd $ORACLE_HOME/dbs
   ```

3. Run the ORAPWD executable to create a new password file.

   ```
   $ORACLE_HOME/bin/orapwd file=orapw<SID> password=your_password entries=10
   ```

   The file name has to be *orapw<SID>*. Entries = 10 is the maximum number of distinct DBAs and OPERs.

   There should be an *orapw<SID>* created in $ORACLE_HOME/dbs

4. Edit the *init<SID>.ora* file. Add or change the following statement `remote_login_passwordfile=exclusive` in the *init<SID>.ora* file.

5. Shut down your database and then restart it.

6. Start Server Manager line mode.
7. Connect as *sys/sys_password*
8. Create a user that will have remote SYSDBA privileges if you do not already have one. In the following example `remote` is the user name which will have SYSDBA privilege.

   ```
   create user remote identified by remote;
   ```

9. Grant connect and SYSDBA privileges to the user.

   ```
   grant connect, resource to remote;
   grant sysdba to remote;
   ```

10. Exit Server Manager Line Mode.

Client Side (Windows NT):

If you want to be able to start up and shut down your target database remotely, follow the steps below:

1. From the Oracle Enterprise Manager Console, select the target database from the Navigator.
2. Start Instance Manager.
3. Log in as a user with the SYSDBA privilege.
4. Double-click Initialization Parameters. The Initialization Parameters Property Sheet appears.
5. Click Save. The Save Configuration property sheet appears.
6. Enter a configuration name.
7. Enter any comments you wish to appear in the Comments column of the Stored Parameters mulit-column list.
8. Click OK.
9. Test the remote SYSDBA login. Start Instance Manager and log in as the user you just created with the SYSDBA privilege. You should be able to startup/shutdown the database using the stored configuration you have created.

Setting Up Backup Modes

If you want to be able to perform on-line backups (hot backups), you must put the database in ARCHIVELOG mode. Otherwise, you must shut down your database; then, perform the backup. By default, the database is in NOARCHIVELOG mode.

To put the database in ARCHIVELOG mode:

1. Edit the following section of the `init.ora` database initialization file according to its instruction:.

```
# Uncommenting the line below will cause automatic archiving if archiving has
# been enabled using ALTER DATABASE ARCHIVELOG.
# log_archive_start = true
# log_archive_dest = %ORACLE_HOME%\database\archive
# log_archive_format = "%%ORACLE_SID%%T%TS%S.ARC"
```

2. Choose Archive Log from the Database menu. Toggle ARCHIVELOG mode on.

For more information on backup and recover strategies, see the *Oracle8 Backup and Recovery Guide*.

Recovery Catalog Setup and Registration for Oracle8

The *recovery catalog* is a repository of information that is used and maintained by Recovery Manager. Recovery Manager uses the information in the recovery catalog to determine how to execute requested backup and restore actions.

To use Oracle8 Recovery Manager with a recovery catalog, you must first install the recovery catalog and then register your database. No setup is required if you are using the control file. It is recommended that you install the recovery catalog on a different location than the target database.

For more information on recovery catalog maintenance, Recovery Manager, and backup and recover strategies, see the *Oracle8 Backup and Recovery Guide.*

Important: Your Oracle Enterprise Manager Job system must be functioning properly before attempting to install the recovery catalog. You will need the Job system to register the target database in the recovery catalog.

Setting Up the Recovery Catalog

To set up a recovery catalog, you must complete the following procedures:

- *Create a Tablespace*
- *Create a User*
- *Run the catrman Script*

Create a Tablespace

1. Start Storage Manager.
2. Log in as a user with the DBA role.
3. Select Create from the Tablespace menu. The Create Tablespace property sheet appears.

> **Note:** You can also use the Navigator menu from the Oracle Enterprise Manager console to bring up the Create Tablespace property sheet.

4. In the Name field, type the name of the new tablespace. For example, `rcvcat`.

5. Click the Add button in the Datafiles section. The Create Datafile property sheet appears, allowing you to specify a new datafile which will belong to the new tablespace.
6. In the Name field, type in complete path and name of the datafile. For example: `c:\orant\database\rcvcat01.dbf`.
7. In the Size section, enter the size of the new datafile. For example, 10 M.
8. Click the OK button in the Create Datafile property sheet.
9. Click the Create button in the Create Tablespace property sheet.

Create a User

1. Start Security Manager.
2. Log in as a user with the DBA role.
3. Choose Create from the User menu. The Create User property sheet appears.

> **Note:** You can also use the Navigator menu from the Oracle Enterprise Manager console to bring up the Create User property sheet.

4. In the General Page, fill in the following information:

   In the Name field, enter the name of the new user. For example, `rman`.

   In the Password and Confirm Password fields, enter a password. For example, `rman`.

   Choose the default tablespace. For example, RCVCAT.

   Choose the temporary tablespace. For example, TEMPORARY_DATA.
5. In the Roles / Privileges Page, fill in the following information:

   Choose Role as the Privilege Type.

   Grant the RECOVERY_CATALOG_OWNER role to the user.
6. In the Quotas Page, specify an unlimited quota for the default tablespace. In this example, the default tablespace is RCVCAT.
7. Click the Create button after specifying the requisite parameters.

Run the catrman Script

1. Start SQL Worksheet.
2. Enter the requisite login information. For example, user=RMAN, password=RMAN.
3. Execute `spool create_rman.log` so that errors will be written in a log file.
4. Execute `catrman` script located in the `Oracle_Home/rdbms/admin` directory.

Registering the Recovery Catalog with Oracle8 Recovery Manager (GUI)

1. Start the Oracle Enterprise Manager Console.
2. From the File menu, choose Preferences. The User Preferences dialog box appears.
3. Make sure the preferred credentials specify SYSDBA access to any target database that needs to be backed up. Normally, the usernames would be those you created in the previous section. Refer to *Setting Up the Recovery Catalog* on page 2-16.
4. Select a target database (version 8.0) from the Navigator.
5. Start Oracle Backup Manager. The Backup Subsystem Selection dialog appears.
6. Select Oracle8 Recovery Manager, then click OK. Backup Manager (using the Oracle8 Recovery Manager subsystem) displays.
7. From the Catalog menu, select Use Recovery Catalog.
8. Enter the requisite login information: username, password, and service. For example, user=RMAN, password=RMAN (or any user you create to access the database with the recovery catalog).
9. You are asked if you wish to register the target database in the recovery catalog. Click Register. At this point, the database registration is sent as a job to the Oracle Enterprise Manager Job system.
10. In Backup Manager, click the registration job in the Active Jobs list to see the current state of the job.
11. Once the job is completed, check the job history in the Jobs property sheet to make sure that the job completed successfully.

When the job completes successfully, your backup and recovery environment is configured. For more information on recovery catalog maintenance, Recovery Manager, and backup and recover strategies, see the *Oracle8 Backup and Recovery Guide*.

Encoding Japanese Characters When Sending Email

A registry key is available for enabling the encoding of Japanese characters when you are sending email.

To set the registry key

1. Run regedit (the Registry Editor).
2. Find the registry key, szDestCharSet, which is located in SOFTWARE\ORACLE\OracleSMPConsole\Email.
3. Double-click the default icon to set the value for szDestCharSet.

Troubleshooting

Please refer to the troubleshooting section below if you should experience any problems during the setup of Oracle Enterprise Manager.

Failed to Authenticate User

"Failed to authenticate user" error occurs when the user submits a job. Use the following hints listed below:

Hint A: If the agent is on a UNIX platform, check $ORACLE_HOME/bin/dbsnmp and make sure that the `dbsnmp` executable is owned by root. Refer to *Verifying that root.sh Had Been Run Successfully* on page 1-11.

Hint B: For an NT node, make sure the preferred credentials for the node is set to be the same as the user that is set up to "Logon as a batch job." Refer to *Setting Preferences* on page 2-12.

Hint C: If an NT agent is started with account other than system, all jobs fail with the message "Failed to authenticate user." This occurs, regardless of whether the job credential account has the "Logon as a batch job" privilege.

Using NT's User Manager from the Administrative Tools program group, you must assign the following privileges (User rights) to the account used to start the agent:

- Log on as service.
- Act as part of the operating system.
- Replace a process level token.
- Increase quotas.

These privileges are required since the NT agent uses CreateProcessAsUser() Win32 API to run jobs. Either create a new account with these privileges or add these privileges to an existing account.

For an example of how to create a new Windows NT user account on the local NT machine and grant privileges to the user, refer to *Creating a New NT User Account* on page 1-4.

For an example of how to assign privileges to an existing local user account, refer to *Assigning Privileges to an Existing NT User Account* on page 1-5.

To start the agent on Windows NT as a non-system account, perform the following steps:

1. Double-click the Services icon in the Control Panel folder.
2. Select the OracleAgent service.
   a. The Startup Type is set to Manual, which allows the agent to be started by a user. If you want the agent to start automatically whenever you start the system, set the Startup Type for Automatic.
   b. Under Log On As section, choose This account (either a local account or domain account).
3. Click the Start push-button to start the agent.

Output from Job Lost

Job fails and the job output window shows:

```
output from job # lost
```

Use the following hints listed below:

- If the agent is on a UNIX platform, the preferred credentials are for a user that has write permissions to the `$ORACLE_HOME/network/agent` directory.
- The `catsnmp.sql` script has not been run on the database. If the agent is running against an NT database, the `catsnmp.sql` is run when you install the database and choose to create a starter database. Any other databases you create yourself will not run this script automatically. Please refer to *Roles and Users Required by the Agent* on page 1-23
- If the agent is running on NT 3.51, the problem has been seen on machines which did not have SP5 installed. Please visit the Microsoft website listed in the readme for information on Service Pack 5.

A job hangs in a "scheduled" state.

Hint A: On NT, the account must have "Logon as a batch job."

If the Oracle partition is formatted as NTFS (NT File System which allows for local security of files and directories) then you should also grant locally the permission set "Full Control" to the user assigned "Logon as a batch job."

To grant the local file permissions:

1. Click Start, Programs, and Windows NT Explorer.
2. Highlight the drive where Oracle is installed and right click.
3. Select Properties-> Security-> Permissions-> Add-> Show Users.
4. Highlight the account which has been granted "Logon as a batch job" and click Add.
5. Click "Type of Access" and select "Full Control" from the drop down list.
6. Then click OK and OK to close the screen.

Hint B: Check to see if the agent can access the Console.

Try pinging the Console from the server machine. If you are not sure about your IP address, please check by running the start button-> settings-> control panel-> network-> protocol-> TCP/IP and clicking the properties button.

If you cannot ping the Console, configure the Daemon Manager host address to the TCP/IP address.

Hint C: Ensure that the DNS Host entry is set to the node name in the listener.ora and tnsnames.ora files.

1. Run the start button-> settings-> control panel-> network-> protocol-> TCP/IP properties.
2. Check the DNS Host entry. For example, make sure that the entry is not set to the name of the previous engineer.

Job Process Crashes

The user running jobs must have read/write permissions to the `$ORACLE_HOME\net80` directory (not just `$ORACLE_HOME\net80\agent` directory) in order to write `*.log` files as well as write permissions to the `$TEMP` directory or the `$ORACLE_HOME` directory.

3

Software Manager Configuration

This chapter discusses configuring the Oracle Software Manager.

Note: Before you can configure the Oracle Software Manager, the Oracle Enterprise Manager must be installed and configured correctly.

Configuring Software Manager

To configure the Oracle Software Manager, you must perform the following procedures:

- modify Node Up / Down event to accept third party events
- register the event with the nodes you wish to manage with Oracle Software Manager
- refresh hosts

Modifying Oracle Host Up/Down Event

To modify the Oracle Host Up / Down Event to accept third party events:

1. Double-click the Enterprise Manager icon from the Oracle Enterprise Manager program group
2. When the Login Information dialog box appears, enter your username, password, and repository service name.
3. In the Event Management Window, click on the Event Set Library tab.
4. Double-click Oracle Host Up / Down.
5. On the Quick Edit Event Set Oracle Host Up Down page, check Accept Third Party Events.

Registering Host Up/Down Event

A *distribution host* is a server that has been designated as a creation, distribution and storage center for software packages and releases. You can designate any host as a distribution host and add or delete distribution hosts from the application, but there must be at least one distribution host designated on each network managed by an Oracle Software Manager application.

Note: In order for Software Manager to designate a distribution host, the Oracle Intelligent Agent must already be installed and configured. Refer to Chapter 1, "Agent Configuration" and to Chapter 2, "Console Configuration" for further information.

To register Host Up / Down event on distribution hosts:

1. Select Register Event Set from Event.
2. Type in Host UpDown as the Set Name.
3. Select all the hosts you want as distribution hosts from the Available Destinations column to add to the Selected Destinations column.
4. Click OK.
5. Click the Registration tab to verify all of the hosts are configured correctly. The hosts should appear under Oracle Host UpDown.
6. After the registration, open Oracle Software Manager by clicking its toolbar icon. You should start the Software Manager console at least once to initialize various state information.

Refreshing Hosts

To view the software already installed on your managed node, you need to refresh hosts:

In the Oracle Software Manager application, select `Refresh Hosts` from the toolbar, menu, or F5.

For more information on using Software Manager, see the *Oracle Enterprise Manager Administrator's Guide*.

Additional Configurations

In addition, the following requirements must be met on the managed nodes in order for Software Manager to function properly.

- If you are on a NTFS file system, the user who runs jobs must have write permission in the agent directory. This user is set as the "Preferred Credential" for that node. On Windows NT with FAT file systems, permission rights is not a issue.
- You must set the environment variable `OSM_PACKAGE_DIR` before starting the agent. The software packages will be stored in this location on agent machines. Depending on the number of packages you create on a host, you may need significant disk space, and you should make sure that the host has ample storage space. If you do not configure this variable, the default will be `%ORACLE_HOME%\NET80\AGENT\PACKAGES` on Windows NT or `%ORACLE_HOME/NETWORK/agent/packages` on UNIX.
- To use Software Manager on UNIX, the `$ORACLE_HOME/network/agent` directory must be writable by the `dba` group.
- For UNIX users only, setting TERM in the login profile is not a requirement. If the TERM value is already set in the login profile, Oracle Software Manager overwrites it with the value VT100 for its purposes.
- The installer must be installed on the Oracle_Home from where the agent is running for the Software Manager agent to work properly.

4

Pack Configuration

Some of the applications in the Oracle Diagnostics Pack and the Oracle Tuning Pack require configuration. This chapter covers the following topics:

These applications are set up after you have installed the Oracle Diagnostics Pack, Oracle Tuning Pack, and the Oracle Change Management Pack.

Note: The Oracle Diagnostics Pack, Oracle Tuning Pack, and the Oracle Change Management Pack are separately licensed and purchased products.

Setting Up the Oracle Diagnostics Pack

The Oracle Diagnostics Pack provides easy-to-use tools for monitoring the health of the system, diagnosing problems, detecting problems automatically, and planning for the future.

This section discusses the following topics:

Setting Up Oracle TopSessions

In order to install the features of Oracle TopSessions,

- you need to create additional tables and views on each database you want to connect to from TopSessions.
- certain SELECT privileges must be granted

The smptsi*xx*.sql scripts have been provided to help automate this process.

The *xx* in the file name identifies the version of the database against which the script should be run. The script for each database version is located in the `$ORACLE_HOME\SYSMAN\ADMIN` directory..

Version of the Database	Script to Run
Oracle 7.3	smptsi73.sql script
Oracle 8.0	smptsi80.sql script

When smptsi*xx*.sql is run, it also automatically runs the following two scripts:

- catblo*xx*.sql
- utlxpl*xx*.sql

These two scripts create in the managed database some additional tables, views, and public synonyms that are required by the Oracle Advanced Events.

To set up Oracle TopSessions for a database:

1. From the Oracle Enterprise Manager program group, double-click on the SQL Worksheet icon to launch the tool.

   For information on starting Oracle Enterprise Manager and tools, see *Starting Enterprise Manager DBA Applications* on page 2-11.

2. Use the Login Information dialog box to connect to the managed database as SYS.

   See the online help or the *Oracle Enterprise Manager Administrator's Guide*.

3. Grant SELECT ANY TABLE privileges to each administrator account. This step may be omitted if the account has already been granted the same privileges as SYSTEM.

   Note that when preparing to run the smptsi*xx*.sql script on managed databases, you should log into each database as SYS, as mentioned previously.

   Refer to *Setting Up the Repository User Account* on page 2-2 for an example of how to grant privileges to an account.

4. Run the smptsi*xx*.sql script for the managed database.

   If the smptsi*xx*.sql script is not run on a managed database, you may see a "Table or View does not exist" message when you try to use Oracle TopSessions.

5. Exit SQL Worksheet.

Setting Up Oracle Trace

Oracle Trace relies on correct client / server configuration.

Oracle Trace requires that:

- the Oracle Enterprise Manager intelligent agent for Oracle 7.3.3 or later is installed and running on the nodes that are targets for Oracle Trace collections
- the Oracle Enterprise Manager Console is running, and the nodes you want to use Oracle Trace on are listed in the console Navigator. Oracle Trace uses the Oracle Enterprise Manager job subsystem to perform its tasks.

This section contains the following topics:

Using Oracle Trace for Oracle 8 Server Collections

If you are using Oracle Trace for Oracle8 server collections, you must set the value of the `ORACLE_TRACE_ENABLE` parameter in your INIT*sid*.ORA file to TRUE.

Using Oracle Trace for Oracle 7.3 Server Collections

If you are using Oracle Trace for Oracle 7.3 server collections, check that the Oracle Trace user account, `TRACESVR`, and the Oracle Trace stored procedure packages: `DBMS_ORACLE_TRACE_AGENT` and `DBMS_ORACLE_TRACE_USER`, exist. If they do not, you must create them by running the `otrcsvr.sql` script as SYS. The `otrcsvr.sql` script is located in `$ORACLE_HOME/otrace/admin` on UNIX systems and in `$ORACLE_HOME\otracexx\admin` on NT systems.

The `otrcsvr.sql` script is run automatically during database installation on most platforms. However, if your server platform is NT, you must run this script manually.

Creating Oracle Trace Repository Tables

All the necessary tables for Oracle Trace are created or updated automatically when any Oracle Diagnostics Pack, Oracle Tuning Pack, or Oracle Change Management Pack product is started for the first time.

Creating Oracle Trace Formatter Tables

If you are using the Oracle Trace Collection Services version 8.0.4, the formatter tables are created for you.

The Oracle Trace formatter tables are required in managed databases by Oracle Trace, which converts and loads an Oracle Trace binary file (*collection_name*.dat) into Oracle tables for access.

Oracle Trace data collected from Oracle 7.3.3 and later databases can only be stored in databases that are using the latest Oracle Trace formatter tables. Use the Repository Manager or the vobsh command from a DOS window to create new formatter tables, delete existing formatter tables, or upgrade earlier versions of formatter tables to the latest version for a database.

Please refer to the *Oracle Enterprise Manager Administrator's Guide* for information on the Repository Manager. The formatter table operations and vobsh commands to perform those operations are described below.

These vobsh commands fail if the specified user does not have an account for the specified service. In this case, create an account for the user on the specified service, then execute the vobsh command again.

Creating New Formatter Tables To create new formatter tables in a database where no formatter tables currently exist, use the vobsh command:

```
vobsh -c "user/password@service" -o CREATE -p "EPCFMT"
```

Note that the formatted data stored in formatter tables can grow very large, so the database in which formatter tables are created should have sufficient space to store large amounts of data.

Dropping Existing Formatter Tables To drop an existing formatter tables from a table, use the vobsh command:

```
vobsh -c "user/password@service" -o DROP -p "EPCFMT"
```

This command drops any version of formatter tables from the specified database. Note that in addition to dropping the existing formatter tables, vobsh also deletes the formatted data stored in the tables.

Upgrading Formatter Tables To validate (upgrade) older versions of the formatter tables (which creates, drops, or upgrades the formatter tables as needed), use `vobsh` command:

```
vobsh -c "user/password@service" -o VALIDATE -p "EPCFMT"
```

This command determines what version of the formatter tables the database has and allows you to upgrade them to the new version, if necessary.

Other Configuration Information

If you experience any problems running Oracle Trace after completing the configuration instructions in this chapter, refer to the "Troubleshooting Oracle Trace" appendix of the *Oracle Enterprise Manager Oracle Trace User's Guide.*

Setting Up Oracle Performance Manager

This section contains the following topics:

Converting Old Performance Manager User-defined Charts

If you created user-defined charts using Oracle Performance Manager 1.5.0 or earlier (the Windows version of Oracle Performance Manager), you can convert those charts so that they can be used with Oracle Performance Manager 1.6.0. To convert the user-defined charts:

1. Run vmmmig.exe, which creates a text file named `vtmusr.txt` in `$ORACLE_HOME\SYSMAN\ADMIN`. This text file contains data about the user-defined charts created using Oracle Performance Manager 1.5.0 or earlier. When you run vmmmig.exe, specify the username, password, and service for the Oracle Enterprise Manager repository that contains the user-defined charts that you want to convert, for example:

   ```
   vmmmig joseph/password@my_rep
   ```

   Note that `my_rep` in the previous command line is the service name for the Oracle Enterprise Manager repository.

2. Run vmm2vtm, which uses the data in `vtmusr.txt` to create user-defined charts in the repository that can be used with Oracle Performance Manager 1.6.0. When you run vmm2vtm, supply a username, password, and service for the Oracle Enterprise Manager 1.6.0 repository and the name of the service under which you want the user-defined charts stored for Oracle Performance Manager 1.6.0, for example:

   ```
   vmm2vtm joseph/password@my_rep my_db
   ```

   Note that `my_rep` in the previous command line is the service name of the Oracle Enterprise Manager 1.6.0 repository and `my_db` is the name of the target service under which you want the user-defined charts stored. In other words, after the previous command is executed, all the user-defined charts are converted and stored under the `my_db` service in the Oracle Performance Manager 1.6.0 tree view.

   If any error messages are generated when you execute vmm2vtm, edit `vtmusr.txt` and execute vmm2vtm again.

Using Performance Manager with Parallel Server

To use Performance Manager in a Parallel Server environment, two instances must be started, otherwise Performance Manager does not treat the server as a Parallel Server.

There are three scenarios for using Performance Manager in an Oracle Parallel Server environment, and different configuration steps for each scenario.

The three scenarios and their configuration steps are:

1. If you are using Performance Manager 1.5.0 or later to monitor an Oracle7 Parallel Server environment, you must run the following scripts:

   - ops_ctab.sql
   - ops_dbl.sql
   - ops_gdl.sql
   - ops_mon.sql
   - ops_pack.sql

   These Parallel Server scripts are provided with the Parallel Server installation. These scripts are also available in the `$ORACLE_HOME\SYSMAN\ADMIN` directory. Running these scripts creates performance monitoring tables and views that Performance Manager requires to fetch and display Oracle7 Parallel Server data in charts.

2. If you are using Performance Manager 1.5.0 to monitor an Oracle8 Parallel Server environment, you must run the `ops_8mon.sql` script.

   This script is provided in the `$ORACLE_HOME\SYSMAN\ADMIN` directory. Running the script creates performance monitoring tables and views that Performance Manager requires to fetch and display Oracle8 Parallel Server data in charts.

3. If you are using Performance Manager 1.5.5 or later to monitor an Oracle8 Parallel Server environment, you do not need to run any of the Parallel Server scripts.

For more information about Parallel Server scripts, see the *Oracle Parallel Server Management User's Guide* in the Parallel Server documentation set.

Setting Up the Oracle Tuning Pack

Oracle Tuning Pack addresses particular tuning needs and activities that ensure the database and applications run at peak efficiency.

This section discusses the following topics:

Setting Up Oracle Tablespace Manager

Before you can use Tablespace Manager's Analyze Wizard against a database, the CHAINED_ROWS table must exist in the database. To create the CHAINED_ROWS table:

1. Log into the database as SYS.
2. Run `utlchain.sql`, which is located in the `$ORACLE_HOME/RDBMS80/ADMIN` directory.

The CHAINED_ROWS table is also used by the Oracle Advanced Events chained rows event.

Setting Up Oracle Expert

Oracle Expert requires a set of database tables in the Oracle Enterprise Manager repository. These tables store the data associated with each Oracle Expert tuning session.

All the necessary tables are created or updated automatically when any Oracle Diagnostics Pack, Oracle Tuning Pack, or Oracle Change Management Pack product is started for the first time.

Setting Up Oracle SQL Analyze

In order to run Oracle SQL Analyze, you must have certain object privileges which are available to users that are granted the DBA role. If you want to assign the minimum privileges required, you can use the optional SQLADMIN role, which assigns the base set of object privileges required by a user.

The VMQROLE.SQL script has been provided to help automate the process of creating the SQLADMIN role. It is located in the `$ORACLE_HOME\SYSMAN\ADMIN` directory.

1. From the Oracle Enterprise Manager program group, double-click on the SQL Worksheet icon to start the tool.
2. Use the Login Information dialog box to connect to the database you want to run SQL Analyze on as SYS.
3. From the Worksheet menu, run the `VMQROLE.SQL` script to create the SQLADMIN role for the managed database.
4. In the SQL Worksheet bottom pane, assign the SQLADMIN role to the user by typing

   ```
   Grant SQLADMIN to <user>;
   ```

5. Exit SQL Worksheet.

A

Configuration Files

This appendix discusses the configuration files that are required for the Oracle Enterprise Manager and its components.

Configuration Files

The Oracle Enterprise Manager Console uses a daemon process for network communication with the Oracle Intelligent Agents on local or remote systems. The network communication is done using Oracle's SQL*Net product (known as Net8 in Oracle Enterprise Manager 1.4 and Oracle8 and up).

Job Scheduling, Event Management, Software Manager, Data Manager, Backup Manager, and Tablespace Manager rely on communication between the Console, agent, and daemon, and require SQL*Net.

Configuration for Console Machine

The following are examples of the configuration files needed on the machine where the Oracle Enterprise Manager Console is run:

sqlnet.ora

```
#################
# Filename......: sqlnet.ora
# Name..........: tcpcom.world
# Date..........: 13-AUG-97 10:09:52
#################
AUTOMATIC_IPC = OFF
TRACE_LEVEL_CLIENT = OFF
SQLNET.EXPIRE_TIME = 0
NAMES.DEFAULT_DOMAIN = world
NAME.DEFAULT_ZONE = world
SQLNET.CRYPTO_SEED = "2418306024240649"
SQLNET.AUTHENTICATION_SERVICES = (NONE)
DAEMON.TRACE_LEVEL = 16
DAEMON.TRACE_DIRECTORY = c:\orant\network\trace
```

Important things to note about this `sqlnet.ora` example is that the domain name is *world*, which means that in this example, any service name in `tnsnames.ora` should have *world* tagged onto it.

Note: "world" is used as an example of a domain name. You may use your own domain name instead of "world."

Also, the three parameters which start with DAEMON control the tracing of the Daemon process. To switch off tracing set DAEMON.TRACE_LEVEL to the value OFF. Also make sure to change the trace directory to reflect the current ORACLE_HOME\NET80\TRACE.

The above example assumes ORACLE_HOME is set to C:\ORANT. When tracing is switched on, a trace file called daemon.trc appears in the directory specified by the DAEMON.TRACE_DIRECTORY parameter. Tracing is useful to set up when debugging the console's operations include discovery, jobs, and events status.

tnsnames.ora

```
#################
# Filename......: tnsnames.ora
# Name..........: LOCAL_REGION.world
# Date..........: 13-AUG-97 10:09:52
#################
mydb.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = tcpcom.world)
          (PROTOCOL = TCP)
          (Host = myhost)
          (Port = 1526)
        )
    )
    (CONNECT_DATA =
       (SID = mysid)
       (GLOBAL_NAME = mydb.world)
    )
  )
```

Note: "mysid" is your SID. "mydb" is the global_name of this database. "world" is an example of a domain name. You may use your own domain name instead of "world." Please refer to the Net8 configuration guide for information about these files.

The standard locations of the `tnsnames.ora file` are: `$ORACLE_HOME/network/admin` or `$ORACLE_HOME/net80/admin`. On Solaris the `tnsnames.ora file` can be located in the `/var/opt/oracle` directory. On other UNIX systems, the `tnsnames.ora file` can be located in the `/etc` directory. On all platforms, you can set a `$TNS_ADMIN` environment variable if you do not want to use any of the default locations.

Configuration for Remotely Managed Machines

The following are some of the example configuration files which may be needed for the machine where the Oracle database and the Oracle Intelligent agent run.

sqlnet.ora

```
#################
# Filename......: sqlnet.ora
# Name..........: myhost.world
# Date..........: 13-AUG-97 10:09:52
#################
AUTOMATIC_IPC = ON
TRACE_LEVEL_CLIENT = OFF
SQLNET.EXPIRE_TIME = 0
NAMES.DEFAULT_DOMAIN = world
NAME.DEFAULT_ZONE = world
SQLNET.CRYPTO_SEED = "2418306024240649"
SQLNET.AUTHENTICATION_SERVICES = (ALL)
```

Important things to note about this `sqlnet.ora` is that the domain name is *world* in this example, which means that in this example, any service name in `tnsnames.ora` should have *world* tagged onto it.

Note: "world" is used as an example of a domain name. You may use your own domain name instead of "world."

---

tnsnames.ora

```
#################
# Filename......: tnsnames.ora
# Name..........: LOCAL_REGION.world
# Date..........: 13-AUG-97 10:09:52
#################
mydb.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = tcpcom.world)
          (PROTOCOL = TCP)
          (Host = myhost)
          (Port = 1526)
        )
    )
    (CONNECT_DATA =
       (SID = mysid)
       (GLOBAL_NAME = mydb.world)
    )
  )
```

listener.ora

```
#################
# Filename......: listener.ora
# Name..........: myhost.world
# Date..........: 13-AUG-97 10:09:52
#################
mylsnr =
  (ADDRESS_LIST =
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= mydb.world)
        )
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= mysid)
        )
        (ADDRESS =
          (COMMUNITY = tcpcom.world)
          (PROTOCOL = TCP)
          (Host = myhost)
          (Port = 1526)
        )
  )
STARTUP_WAIT_TIME_mylsnr = 0
CONNECT_TIMEOUT_mylsnr = 10
TRACE_LEVEL_mylsnr = OFF
SID_LIST_mylsnr =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = mysid)
      (ORACLE_HOME = /myoraclehome)
    )
  )
```

This example of `listener.ora` defines the listening address for the SQL*Net Listener *`mylsnr`* and tells it about the Oracle Database *`mysid`*. To start this listener on UNIX, enter the command:

```
$ lsnrctl start mylsnr
```

This command explicitly specifies the name of the SQL*Net listener. If you are using the default listener name, you do not need to specify the name when starting the listening process. (The default listener name is "listener.") However, in this

example, the listener's name has been changed to "mylsnr." Since this is not the default name, it must be specified during startup.

snmp.ora for Pre-7.3.3 Agents Only

```
#################
# Filename......: snmp.ora.sample
#################
snmp.visibleservices = (mydb_name.world, myhost_name_mylsnr.world)
snmp.index.mydb_name.world = 1
snmp.index.myhost_name_mylsnr.world = 2
snmp.contact.mydb_name.world = contact_info
snmp.contact.myhost_name_mylsnr.world = contact_info
snmp.sid.mydb_name.world = server_id
snmp.oraclehome.mydb_name.world=$ORACLE_HOME
nmi.register_with_names=false
nmi.trace_level = 0
nmi.trace_directory = $ORACLE_HOME\network\trace
dbsnmp.address = (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=myhost_
name)(PORT=1748)))
dbsnmp.spawnaddress = (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=myhost_
name)(PORT=1754)))
```

This `snmp.ora` example is used by a pre-7.3.3 Intelligent Agent. Pre-7.3.3 Intelligent Agents can use any unused port number, as long as the numbers match the `tnsnames.ora` entries for the Agent. `snmp.ora` should be created by the Oracle Network Manager or Topology Generator.

The configuration files, `snmp_ro.ora`, and `snmp_rw.ora`, provide configuration parameters for the 7.3.3 (and later) agent. These files are created automatically by the Intelligent Agent.

snmp_ro.ora

The `snmp_ro.ora` file is located in `$ORACLE_HOME\net80\admin` on Windows NT platforms. This file is located in `$ORACLE_HOME/network/admin` on UNIX. Do not update this read-only file. It contains the following parameters:

```
SNMP.VISIBLESERVICES = (LISTENER, service_name1, service_name2, ...)
SNMP.SID.service_name = server_id
SNMP.ORACLEHOME.service_name = ORACLE_HOME_DIR
```

snmp_rw.ora

The `snmp_rw.ora` is located in `$ORACLE_HOME\net80\admin` on Windows NT platforms. You can modify this read-write file, but this should be done carefully. It contains the following parameters:

```
SNMP.INDEX.service_name = unique_index_number
SNMP.CONTACT.service_name.world = "contact_info"
NMI.REGISTER_WITH_NAMES = FALSE
NMI.TRACE_LEVEL = OFF | USER | ADMIN | nn
```

Agents, versions 7.3.3 and later, require port address 1748 and 1754. TCP/IP protocol is required to automatically discover services with the agent. The port address is automatically set.

The following parameters are not automatically generated, but may be added to the file:

```
SNMP.CONNECT.service_name.USER = user_name
SNMP.CONNECT.service_name.PASSWORD = password
SNMP.DBPOLLTIME = nn
DBSNMP.IPCTIME = nn
NMI.TRACE_DIRECTORY = directory
NMI.TRACE_FILE = filename
NMI.LOG_DIRECTORY = directory
NMI.LOG_FILE = filename
```

services.ora

The `services.ora` file is created when the agent starts and is located in `$ORACLE_HOME\net80\agent` on the Windows NT platform and `$ORACLE_HOME/network/agent` on UNIX. This file contains a list of the services, such as Oracle databases and listeners, on the node where the agent resides. This file is retrieved from the agent by Oracle Enterprise Manager through the Navigator Discovery menu options.

Note: Do not manually edit the services.ora file. The agent rewrites the file on startup.

Parameters for snmp*.ora Files

These parameters are used in the `snmp_ro.ora` and `snmp_rw.ora` configuration files for the Intelligent Agent release. These parameters are also used in the `snmp.ora` file, the primary configuration file for Intelligent Agent releases prior to the 7.3.3 release.

In the following parameters, note these substitutions:

- *service_name* is the name of the service, database or listener, that you intend to monitor as it appears in `tnsnames.ora`.
- *host_name* is the host name of your machine.
- *world* is the name, if any, of your community.

Note: If *.world* is used in the `sqlnet.ora` file, then you must use *.world* in the `snmp.ora` and `tnsnames.ora` files. For example, *service_name.world* and *host_name_lsnr.world*.

SNMP.VISIBLESERVICES = (service_name1.world, service_name2.world, ...)
The name of the services that the agent is monitoring. Each database and each SNMP-manageable service must be listed. For a database, the service name is the name of the database as it appears in the `tnsnames.ora` file or in the Names Server.

SNMP.INDEX.service_name.world = index_number
The unique index number of the service that the agent is monitoring. The index number can be any number. The only limitation is that if you have more than one index line, the index numbers must be unique. For example:

```
snmp.index.<service_name1>=10
snmp.index.<service_name2>=20
```

SNMP.SID.service_name.world = server_id
The server Id (SID) of the database service that the agent is monitoring.

SNMP.CONNECT.service_name.world.USER = user_name
The username that the subagent uses to connect to the database. The default is `dbsnmp`. This parameter is optional. The catsnmp.sql script should be edited and reexecuted if this parameter is not the default setting.

The "subagent" refers to the Intelligent Agent. Sometimes, the Intelligent Agent is called a subagent to the master SNMP agent when configuring SNMP on a server. However, SNMP does not have to be configured on the server before the Intelligent Agent will work (except for the Netware platform). For security reasons, the customers sometimes do not want to use the default Intelligent Agent database account / password of dbsnmp / dbsnmp. The example listed should only be used if they want to change the Intelligent Agent's database logon account.

SNMP.CONNECT.service_name.world.PASSWORD = password
The password for the username that is used by the subagent to connect to the database. The default is `dbsnmp`. This parameter is optional. The catsnmp.sql script should be edited and reexecuted if this parameter is not the default setting.

The "subagent" refers to the Intelligent Agent. Sometimes, the Intelligent Agent is called a subagent to the master SNMP agent when configuring SNMP on a server. However, SNMP does not have to be configured on the server before the Intelligent Agent will work (except for the Netware platform). For security reasons, the customers sometimes do not want to use the default Intelligent Agent database account / password of dbsnmp / dbsnmp. The example listed should only be used if they want to change the Intelligent Agent's database logon account.

SNMP.ORACLEHOME.service_name.world = *ORACLE_HOME_DIR*
The Oracle home directory of the database. A separate entry is required for each database even if `ORACLE_HOME_DIR` is the same for all services.

SNMP.CONTACT.service_name.world ="contact_info"
A string containing contact information, such as name, phone number, and email, of the administrator responsible for the service. This parameter is optional.

DBSNMP.POLLTIME = nn
The time interval (seconds) that the agent polls the database to check whether it is down. If the database has gone down or was never connected, this is the interval between retries. The default is 30 seconds.

DBSNMP.IPCTIME = nn
The time interval (seconds) that the agent's Work process pings its Comm process to check whether it is down. If Comm is unable to respond to Work's ping within this time, Work will kill the old Comm process and spawn a new one. Users can lengthen the interval if they want to be able to run long events, separately from the database-checking interval. The default is 30 seconds.

This parameter is not pertain to the NT port of the agent.

NMI.TRACE_LEVEL = OFF | USER | ADMIN | nn
Turns on tracing at the specified level. Oracle recommends that you set the trace level to 13. Level 16 produces a deluge of information, which is only useful if a bug is being investigated. With a level of 16, you can see actual TCP/IP packet contents. With a level of 15, I can only see that packets are being passed. This parameter is optional.

NMI.TRACE_DIRECTORY = *directory*
Directory where trace file is written. The setting is only relevant in conjunction with `nmi.trace_level`. If omitted, trace files are written to `$ORACLE_HOME\network\trace`. This parameter is optional.

NMI.TRACE_FILE = *filename*
Filename of the trace file. This parameter is optional.

NMI.LOG_DIRECTORY = *directory*
Directory where log file is written. This parameter is optional.

NMI.LOG_FILE = *filename*
Filename of the log file. This parameter is optional. On Windows NT, the filename defaults to `dbsnmp`.

Note: The following addresses are automatically set by the agent. Changing the addresses makes the agent undetectable by the Enterprise Manager Console and forces a manual configuration setup.

dbsnmp.address =(ADDRESS=(PROTOCOL=protocol) (HOST=host_ name)(PORT=port_no)))

The TNS address that the agent uses to listen for incoming requests. There should be no space or return characters in the address. This parameter is the address that the Agent listens on for network connections.

TCP/IP must be installed on the server since it is required to automatically discover services with the agent.

The agent requires PORT=1748. The port address 1748 is a registered TCP port granted to Oracle by the Internet Assigned Number Authority (IANA). The port address is automatically set. Changing this port makes the agent undetectable by the Enterprise Manager Console and forces a manual configuration setup.

For agent releases previous to the 7.3.3 release, this address must match exactly the entry for this agent in the tnsnames.ora file on the machine where the Oracle Enterprise Manager Console resides.

dbsnmp.spawnaddress =(ADDRESS= (PROTOCOL=protocol) (HOST=host_ name)(PORT=spnport_no)))

The TNS address which the agent can use to accept RPC's. This address is used for file transfers. The spnport_no used in this parameter is different than port_no used in the DBSNMP.ADDRESS parameter.

The agent PORT=1754. The port address 1754 is a registered TCP port granted to Oracle by the Internet Assigned Number Authority (IANA). Changing this port makes the agent undetectable by the Enterprise Manager Console and forces a manual configuration setup.

B

Directory Structure

This appendix describes the directory structure of Oracle Enterprise Manager components and the Intelligent Agent. The entire directory structure is not detailed; only the major sections.

Directory Structure of Oracle Enterprise Manager

The following directories and files are used by Oracle Enterprise Manager components. Software and script files are installed in the ORACLE_HOME directory of the PC where Enterprise Manager is installed. The examples are for a Windows NT platform and the default ORACLE_HOME directory for Windows NT, \orant, is used.

\orant\bin
Enterprise Manager executable and DLL's are located in this directory. Help files are also located here.

\orant\dbs
Enterprise Manager message files are here.

\orant\otrace...
Files associated with Oracle Trace are located here.

Enterprise Manager Components

The Oracle Enterprise Manager components are installed in the `\orant\sysman\...` directories. These directories include:

\orant\sysman\admin
SQL scripts used to set up Oracle Enterprise Manager components are located in this directory. In addition, bitmaps that are displayed in the product are stored here.

\orant\sysman\admin\rdbms70, rdbms71, rdbms72, rdbms73, rdbms80
SQL scripts required for TopSessions setup are located in these directories. The scripts are specific to an Oracle database version. These scripts are not directly accessed by the user and should not be run manually.

\orant\sysman\admin\tcl
tcl files are located in this directory.

\orant\sysman\bmp
Sample bitmaps that are used for map backgrounds are installed here.

\orant\sysman\doc
Enterprise Manager online documentation is located in subdirectories. The files are in HTML format with a `.htm` extension. The `toc.htm` file contains the table of contents for each book and should be viewed first.

\orant\sysman\expert
Oracle Expert sample files are located here.

\orant\sysman\ifiles
The Instance Manager stores imported configuration files here.

\orant\sysman\ocm155
Change manager files are located in this directory.

\orant\sysman\ole2
NT registry configuration for Enterprise Manager console, daemon, and applications are located in this directory.

\orant\sysman\osm10
Files and directories used by Software Manager are located here.

\orant\sysman\output
Job output files are located in this directory.

\orant\sysman\qtour
Quick tour files for Expert and the Console are located in this directory.

\orant\sysman\scripts
DBA, SQL, and Tcl scripts used in the Job system are located in subdirectories in this directory.

\orant\sysman\sdk
The sample files and scripts for the Enterprise Manager Software Developer's Kit (SDK) are located in subdirectories in this directory.

\orant\sysman\temp
Temporary files (such as map state) are located in this directory.

\orant\sysman\vtm155
Performance Manager files are located in this directory.

\orant\sysman\vtp155
Capacity planner files are located in this directory.

Warning: Do not delete any files from the `temp` directory.

Network Files

The Oracle Enterprise Manager network configuration files and trace files are stored in the `\orant\net80\...` directories. These directories include:

\orant\net80\admin
The current SQL*Net configuration files and default event definition are located in this directory.

\orant\net80\log
The client and sever log files are usually written here.

\orant\net80\trace
The sever and client trace files are usually written here.

Directory Structure of the Intelligent Agent

The following directories and files are used by the agent on a node. All the subdirectories are located under the ORACLE_HOME directory. The examples are for a Windows NT platform and the default ORACLE_HOME directory for Windows NT, \orant, is used.

\orant\agentbin
This is the location of the agent executable file, dbsnmp.

\orant\net80\admin
The is the location of the configuration files for the agent, such as snmp_ro.ora and snmp_rw.ora.

\orant\net80\log
The nmiconf.log file is located here. This file contains errors that occur when the agent is discovering services.

\orant\net80\mesg
This is the location of the agent message file.

\orant\net80\agent
This is the location of the *.q files that contain information on the current jobs and events that the agent is managing for the node. The services.ora file is also located here. The user whose credentials are used to run a job with an output file should have write permissions in this directory.

\orant\rdbms80\admin
This is the location of SQL scripts that create necessary database accounts and roles used by the agent. The catsnmp.sql script creates the dbsnmp account and the SNMPAGENT role. The catsnmp.sql script is run by catalog.sql during the installation of a database. The catnsnmp.sql script removes the role and user installed by the catsnmp script.

\orant\net80\agent\config
This is the location of the agent tcl scripts.

\orant\net80\agent\doc
This is the location of the agent readme file.

\orant\net80\agent\events\oracle
The oracle event scripts are located under this directory.

\orant\net80\agent\jobs\oracle
The oracle job scripts are located under this directory.

\orant\net80\agent\mibs
This is the location of the mib initialization file, `orainit.mib`.

\orant\net80\agent\packages
This is the location of the Oracle Software Manager files.

\orant\net80\agent\tcl
This is the location of Tcl initialization scripts.

C

Hints and Tips

Please check the compatibility matrix in the Oracle Enterprise Manager Readme before installing or using Enterprise Manager. The `readme` is located in the `$ORACLE_HOME\SYSMAN\ADMIN` directory.

Oracle Enterprise Manager 1.x.x can only be installed in the same ORACLE_HOME as an NT 8.x.x database because Oracle Enterprise Manager 1.x.x. and the 8.x.x database share required support files. Oracle Enterprise Manager 1.x.x can be used to administer other database versions but cannot be installed in the same ORACLE_HOME.

This appendix describes several hints and tips which enable the use of Oracle Enterprise Manager, its components, and the Intelligent Agent.

Troubleshooting

The following are possible troubleshooting issues.

System Registry Errors

If a message alerts you that the Oracle Installer was unable to register Enterprise Manager components or you have trouble running any of the programs, you must register the components manually after exiting the Oracle Installer. Use the Windows File Manager or Explorer to locate the `ORACLE_HOME\SYSMAN\OLE2` directory, then double-click on each of the `.REG` files to register each component.

If you have problems with the parameters or task property sheets when creating a job with the Job Scheduling system, the `.OCX` files may have failed to register properly during registration. To register the `.OCX` files, perform the following steps:

1. Open a DOS window and change to the `\ORACLE_HOME\BIN` directory.
2. Enter `DIR *.OCX` to list the all `.OCX` files.
3. Enter `REGSVR32 name.OCX` for each `.OCX`, where `name` corresponds to each of `.OCX` filenames.

   For example: `regsvr32 vojt.ocx`

Job and Event Systems

If you are having trouble using the Job or Event systems of the Console, check the Daemon and Agent trace file and look for error messages saying the Daemon could not resolve an agent address into a host name. On a Windows NT platform, you can also check the Event viewer in Administrative Tools.

Intelligent Agent

If the Intelligent Agent does not start, check one of the following areas for information:

- `ORACLE_HOME/network/log/dbsnmp*.log` file for errors on UNIX
- Select Application from the Event Viewer on Windows NT.

For detailed information about troubleshooting the agent, please refer to Chapter 1, "Agent Configuration".

Error Messages When Starting the NT Intelligent Agent

If you see an OS error when starting the agent, check to see if it is actually an agent error as described in snmimsg.mc. If you do not receive a print out with the cause of the error, use the Event Viewer in the Administrative tools group of Windows NT. You should find the true cause of the problem documented.

These messages may be incorrectly identified as OS errors.

Index	Description
01	OracleAgent failed to register its Service Control Handler.
02	OracleAgent failed to report its status to the Service Control Manager.
03	OracleAgent failed to create a thread synchronization object.
04	OracleAgent failed to create a thread.
05	OracleAgent failed to allocate memory.
06	OracleAgent failed to get the encryption key.
07	OracleAgent failed to run auto-discovery script nmiconf.tcl. Look in the nmiconf.log, for more information.
08	OracleAgent failed to initialize Oracle CORE library.
09	OracleAgent failed to initialize Oracle NLS library.
10	OracleAgent failed to initialize Oracle SQL*Net library : %1.
11	OracleAgent failed to initialize DES encryption.
12	OracleAgent failed to initialize Oracle Remote Operations library.
13	OracleAgent failed to create package index for Oracle Software Manager.
14	OracleAgent failed to create file dbsnmp.ver.
15	OracleAgent failed to create / read queue file.
16	OracleAgent failed to create job scheduling symbol table.
17	OracleAgent failed to initialize its connection cache.
18	OracleAgent failed to signon to SNMP master agent.
19	OracleAgent failed to read SNMP indexes from parameter file.
20	OracleAgent failed to connect to database.
21	OracleAgent failed to build SNMP cache.
22	OracleAgent failed to build MIB.

Index	Description
23	OracleAgent failed to register MIB row.
24	OracleAgent failed to restart its communication thread.

Testing the Connectivity to Any SID

To test the connectivity to any SID, you must configure the `tnsnames.ora` to have a sqlnet connect string to connect to that particular SID, then use sql*plus to connect to that connect string. Use Net8 Assistant for Net8 and Oracle Network Manager / Sqlnet easy config for sqlnet 2.x.

An example is shown below of a sqlnet connect string *scott.world* that will connect to the SID that has the name *ORCL* found on *scott-PC* node through TCP / IP.

```
scott.world=
        (Description=
              (Address_list=
                          (Address=
                          (community=TCP.world)
                          (protocol=TCP)
                          (host=scott-PC)
                          (port=1521)
               )
              (Address=
                          (community=TCP.world)
                          (protocol=TCP)
                          (host=scott-PC)
                          (port=1526)
               )
             )
             (connect_data=
                    (SID=ORCL)
             )
          )
```

In sql*plus type:

```
sqlplus> connect username/password@scott
```

sql*plus should return the following:

```
sqlplus>connected.
```

If sql*plus returns any errors, please refer to the *Sqlnet Configuration Guide* for further details.

Manually Defining a Local Database

If your agent is not running, you can manually define a local database. This procedure must be performed before you can use any of the basic applications or performance pack applications against the local database.

The steps can be done on Windows95 using Personal Oracle 7. The procedure is similar for Windows NT.

1. Edit `\orawin95\net80\admin\tnsnames.ora` and define a local database.

   In the example, `local.world` is used. The definition in the `tnsnames.ora` file should look like the following:

```
local.world =
(DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = beq.world)
          (PROTOCOL = BEQ)
          (PROGRAM = oracle73)
          (ARGV0 = oracle73ORCL)
          (ARGS =
'(DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))')
      )
            )
        (CONNECT_DATA = (SID = ORCL)
        )
  )
```

2. Start Oracle Enterprise Manager and login to the repository.
3. Under Navigator-->Service Discovery-->Manually Define Services, click the Next button, then enter the node name of the PC at the node name prompt. Then click the Add button.
4. Move down to the databases: table (under the node name section) and enter the name of the database alias as defined in `tnsnames.ora`. In the example, the alias is `local.world`. Click the Next button, then click Finish.
5. From the Console window, expand the Databases folder, and double click the newly added database to confirm that the connection is working properly.

Manually Creating, Dropping, or Upgrading the Repository

If the automatic repository create or upgrade (validate) operation fails, you can use the Repository Manager (It is available under the Start-->Programs-->Oracle Enterprise Manager folder) or run the operation from the DOS command line. Please refer to the *Oracle Enterprise Manager Administrator's Guid*e for detailed information about the Repository Manager.

You can also drop a repository from the DOS command line.

1. Open a DOS window and change to the `\ORACLE_HOME\BIN` directory.
2. At the DOS prompt, enter `vobsh` followed by one or more of the command line switches in Table C–1, "Command Line Arguments for vobsh".

   For example, the following command validates and, if necessary, upgrades the Enterprise Manager repository.

   ```
   vobsh -c "scott/tiger@mydb" -o VALIDATE -p "Enterprise Manager"
   ```

Command line arguments are listed in the table below.

Table C–1 Command Line Arguments for vobsh

Arguments	Description
-b	Sets batch mode. All prompting will be disabled.
-c cn	Sets the connect string. The parameter cn must be a quoted string containing the user / password specification, such as "scott / tiger@mydb".
-h	Displays the command line options.
-o *option*	Sets the repository manager option. Valid options are CREATE, DROP, and VALIDATE the repository.
-p *name*	Sets the product name and identifies the specific repository. The name must be a valid sub-component name or a valid group name. Valid sub-component names are Enterprise Manager, Software Manager, Oracle Expert, and Oracle Trace. Valid group names are: ALL, CONSOLE, PERFORMANCE. ALL and PERFORMANCE include all sub-component repositories. CONSOLE includes the Enterprise Manager and Software Manager repositories. The *name* parameter must be in quotes.

Using Online Help

Enterprise Manager uses online help to provide additional information on the entire product and its optional components. To display context-sensitive help on the Console, perform the following steps:

1. Move the mouse pointer to your specific area of interest.
2. Press the F1 key.

If you want help on a particular dialog box, press its Help button. You can also access the help system from the Help menu on the Console. The Contents page lists the major help topics by title. The Index page lists topics by key words. The Find page is a Windows user option that can be set up with a wizard. The setup is performed the first time you access the page.

Glossary

agent

See *Oracle Intelligent Agent*.

agent discovery

The agent discovers all services on its node.

access (database)

One of the three standard privileges to an Oracle database. A DBA can grant a user one or more of the database privileges.

account

An authorized user of an operating system or a product. Depending on the operating system, may be referred to as ID, User ID, login, and others. Accounts are often created and controlled by a system administrator.

application

One or more program modules used to achieve a specific result. applications can be nested within other applications.

authenticate

The process of verifying the identity of a user, device, or other entity in a computer system, often as a prerequisite to allowing access to resources in the system.

authorization

Permission given to a user, program, or process to access resources in the system.

back up

The creation of a copy (or copies) of existing data to enable recovery of that data subsequently required.

batch

Users specify a query to run in batch if they do not want to stop work to wait for the results. The computer decides when to run the query. Users can go back and look at the results of the query later.

cache (memory)

A temporary storage place for database data that is currently being accessed or changed by users or for data that Oracle Server requires to support users. The term buffers (database) is used interchangeably with cache.

click

To press and quickly release the left mouse button. You generally use the click function to highlight items in a window. You click and hold down the left mouse button primarily to make a choice within a choice. For example, you may click and hold on a menu option to bring up a submenu, and then release it after making a selection from the submenu.

client

A user, software application, or computer that requests the service, data, or processing of another application or computer (the "server"). In a two-task environment, the client is the user process. In a network environment, the client is the local user process and the server may be local or remote.

Communication Daemon

A process that runs on the same computer as the OEM Console. The process sends and receives information to and from the OEM Console and the Oracle Intelligent Agent.

connect

To access an Oracle database using a valid username and password. Similar to logging on to an operating system using an account name and password. You must connect if you want to create or modify queries or access a display stored in a database.

Console

1. Graphical user interface (GUI) that provides menus, toolbars, launch palettes, and the framework to allow access to Oracle tools, plus utilities available through other vendors. 2. A computer running either Windows NT or Windows 95 operating system where Oracle Enterprise Manager is installed. The console is also referred to as the OEM Console.

create

To bring an object (a table, block, field and others) into existence, usually by defining its primary characteristics.

Daemon Manager

Allows you to manage the Console's communication daemon.

database

1. A set of operating system files, treated as a unit, in which an Oracle Server stores a set of data dictionary tables and user tables. A database requires three types of files: database files, redo log files, and control files. 2. The disk space corresponding to this set of files. 3. A subset of the database objects necessary to support a single database application.

database administrator (DBA)

1. A person responsible for the operation and maintenance of an Oracle Server or a database application. the database administrator monitors its use in order to customize it to meet the needs of the local community of users. 2. An Oracle username that has been given DBA privileges and can perform database administrative functions. Usually the two meanings coincide. There may be more than one DBA per site.

database name

A unique name assigned to a connect descriptor in a TNSNAMES.ORA file. The use of database names allows connect descriptors to be easily referenced.

data definition language (DDL)

The category of SQL statements that define or delete database objects such as tables or views. Examples are the CREATE, ALTER, and DROP statements.

DLL

Dynamically linked library on a Windows NT/95 platform.

datafile

A file which contains all the database data. A datafile can be associated with only one database.

Data Manager

A tool which allows you to transfer data in and out of an Oracle database.

default

A value supplied by the system when a user does not specify a required command parameter or qualifier.

discovery

The process of identifying services and adding them to the Oracle Enterprise Manager Navigator.

discovery cache

An in-memory list of information about services and nodes.

distribution host

A server that has been designated as a creation, distribution and storage center for software packages and releases.

event

An occurrence associated with a service monitored by the Intelligent Agent. Users select the events they want to monitor for and register them with the Intelligent Agent.

Event Management System

Allows you to monitor specific event conditions that occur in your network environment.

export

To write database data to files for the purpose of archiving or moving data between operating systems or Oracle databases. To store a copy of a module to a file or database.

extent

A logical unit of database storage space allocation made up of a number of contiguous data blocks.

file

A collection of data treated as a unit, such as a list, document, index, note, set of procedures, and so on. Generally used to refer to data stored on disks.

filename

The name component of a file specification. A filename is assigned by either the user or the system when the file itself is created.

fixit job

An unscheduled job associated with a registered event by an Oracle Enterprise user. When the event the fixit job is associated with occurs, the job executes automatically to resolve the problem.

grant

To give a user access to modules. Only a module's creator can grant its access to other users.

group

A user-definable collection of similar objects such as databases, listeners, Names Servers, or nodes that share a common location or function.

index

Optional structures associated with tables and clusters which provide a fast access path to the data.

instance

Combination of the background processes and memory buffers. An Oracle instance has two types of processes: user processes and Oracle Processes.

integrity constraint

A declarative method of defining a rule for a column of a table.

Navigator

Provides a tree list of all the objects in a network, providing a direct view of objects such as user-defined groups, nodes, listeners, names servers, and databases, plus the objects that they contain.

node

A uniquely addressable computer attached to a network.

Oracle Intelligent Agent

A process that interacts with the OEM Console's Communication Daemon. The Agent receives and processes requests from the Communication Daemon over SQLNET and RPC. The Agent can also communicate through SNMP to an SNMP Master Agent allowing third party SNMP Monitoring stations to access information about Oracle through Management Information Blocks (MIB).

Oracle Process

A server process that performs work for user processes and background processes that perform maintenance work for the Oracle Server.

OEM Repository

A set of tables in an Oracle database which stores data and information required by the OEM Console. The tables are owned by the account that creates them.

Oracle Software Manager (OSM)

A tool that will allow installation of OSM compliant software remotely to other Windows NT servers or Windows clients.

Oracle Performance Pack

A set of valuable Oracle tools integrated with the OEM Console for monitoring and maintaining performance of Oracle databases. The Performance Pack is purchased separately from Oracle Enterprise Manager.

OraTcl

Oracle's extensions to Tcl, including Oracle database-specific commands.

package

Provides a method of encapsulating and storing related procedures, functions, and other package constructs together as a unit in the database.

package body

Defines all constructs, public and private, of the package.

package specification

Declares all public constructs of the package.

procedure

A set of SQL and PL/SQL statements grouped together to perform a specific task.

property sheet

A dialog box used to specify options when creating or altering a composite instance, such as user or tablespace.

Q files

The queues are required for event and job scheduling. They are created the first time the agent starts up.

Q file	Description
ereg.q	Event Registeration queue, a list of events that have been successfully registered on the console.
user.q	User queue, contains list of all registered users.
job.q	Job request queue, list if all successfully scheduled jobs. Canceling jobs removes them from this queue.
jstat1.q	Job result queue, results stored before successfully sent back to the Daemon.
evocc1.q	Event occurence queue, results stored before successfully sent back to the Daemon.

redo log

A set of files that protect altered database data in memory that has not been written to the datafiles.

repository

A set of tables in an Oracle database.

role

Named groups of related privileges that are granted to users or other roles.

rollback segment

A portion of the database that records the actions of transactions if the transaction should be rolled back (undone). Rollback segments are used to provide read consistency, to rollback transactions, and to recover the database.

scalability

A measure of how well the software or hardware product is able to adapt to future business needs.

schema

A named collection of schema objects, such as tables, views, clusters, procedures, and packages.

schema object

Logical structures that directly refer to the database's data. Schema objects include such structures as tables, views, sequences, stored procedures, synonyms, indexes, clusters, and database links.

sequence

A database object used to generate a serial list of unique numbers for numeric columns of a database's tables.

session

A specific connection of a user to an Oracle instance via a user process.

Server

A computer running either UNIX or Windows NT operating system where Oracle database is installed.

Simple Network Management Protocol (SNMP)

A protocol which allows administrators to manage heterogeneous systems with a single administration interface.

snapshot log

A table associated with the master table of a snapshot.

snapshot

A read-only copy of a master table located on a remote node. It is periodically refreshed to reflect changes made to the master table.

Software Manager

A tool which allows you to distribute, install, and uninstall software packages on servers and clients throughout the network.

SQL Worksheet

A window to enter, edit, and execute SQL and PL/SQL code. It maintains a history of issued commands for easy retrieval.

synonym

An alias for a table, view, snapshot, sequence, procedure, function, or package.

System Global Area (SGA)

An area of memory used for database information shared by the database users.

tablespace

One or more logical storage units into which a database is divided. A tablespace is used to group related logical structures together.

Tool Command Language (Tcl)

An industry standard string processing language for issuing commands to interactive programs. Jobs and events are Tcl scripts processed by the Intelligent Agent.

topology

A description of the interconnections and relationships in a network. Some versions of Oracle Enterprise Manager use a topology file to populate the Enterprise Manager Navigator with manageable entities.

trigger

A stored PL/SQL procedure that is implicitly started when an INSERT, UPDATE, or DELETE statement is issued against the associated table.

user process

A process that executes the code of an application program or an Oracle tool.

view

A custom-tailored presentation of the data in one or more tables. Also known as a *stored query.*

Index

C

D

E

G

H

S

T

U

V

W