

**Oracle® Data Mining**

Administrator's Guide

10g Release 2 (10.2)

**B14338-01**

June 2005

Copyright © 2005, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

---

---

# Contents

|   |      |
|---|------|
| <b>Preface</b> .....  | vii  |
| Audience .....  | vii  |
| Documentation Accessibility .....                                   | vii  |
| Related Documents .....   | viii |
| Conventions .....   | viii |
| <br>  |      |
| <b>1 Installing Oracle Database for Data Mining</b>                 |      |
| <b>Who Needs to Install Oracle Database?</b> .....                  | 1-1  |
| <b>Installation of Oracle Database 10g</b> .....                    | 1-2  |
| Hardware Requirements .....   | 1-2  |
| Software Requirements .....   | 1-2  |
| Preliminary Steps .....   | 1-3  |
| Installing Oracle Database .....                                    | 1-3  |
| Installing the Data Mining Sample Programs .....                    | 1-7  |
| <b>Tips for Linux Installations</b> .....                           | 1-8  |
| <b>Additional Information for Enterprise Installations</b> .....    | 1-8  |
| Creating Data Mining Tablespaces on Real Application Clusters ..... | 1-8  |
| Installing the Data Mining Scoring Engine .....                     | 1-9  |
| <b>Verifying Installation of Data Mining Components</b> .....       | 1-9  |
| <b>Upgrading from Oracle Database 10g Release 1 (10.1)</b> .....    | 1-10 |
| Post-Upgrade Steps for Data Mining .....                            | 1-10 |
| Upgrading PL/SQL Models .....                                       | 1-11 |
| Downgrading Oracle Data Mining .....                                | 1-12 |
| <br>  |      |
| <b>2 Administering Oracle Database for Data Mining</b>              |      |
| <b>Introduction to Oracle Administration Tools</b> .....            | 2-1  |
| Local Administration on Microsoft Windows .....                     | 2-1  |
| Oracle Enterprise Manager Database Control .....                    | 2-1  |
| SQL*Plus .....  | 2-2  |
| Database Configuration Assistant .....                              | 2-2  |
| Oracle Universal Installer .....                                    | 2-3  |
| Oracle Services .....   | 2-3  |
| Local Administration on Linux .....                                 | 2-3  |
| Remote Administration .....   | 2-4  |

|   |      |
|---|------|
| <b>Creating Oracle Database Users for Data Mining</b> ..... | 2-4  |
| Creating a Demo User for Data Mining .....                  | 2-4  |
| Creating Analytic Users for Data Mining .....               | 2-4  |
| Creating Default Tablespaces .....                          | 2-5  |
| Creating Database Users for Data Mining .....               | 2-5  |
| Granting Access Rights .....                                | 2-5  |
| What the dmshgrants Script Does .....                       | 2-5  |
| What the dmshgrants Script Does Not Do .....                | 2-6  |
| About the DMSYS Schema .....                                | 2-6  |
| <b>Creating a Sample Schema for Data Mining</b> .....       | 2-6  |
| <b>Exporting and Importing Data Mining Models</b> .....     | 2-6  |
| Prerequisites .....   | 2-7  |
| Directory Objects .....                                     | 2-7  |
| Additional Database Privileges .....                        | 2-8  |
| PL/SQL APIs for Exporting and Importing Models .....        | 2-8  |
| Java APIs for Exporting and Importing Models .....          | 2-8  |
| Tables Created By Exporting and Importing Models .....      | 2-8  |
| Example: Exporting and Importing Models .....               | 2-9  |
| Exporting Models from the DMUSER Schema .....               | 2-9  |
| Importing Models Into the Same Schema .....                 | 2-10 |
| Importing Models Into a Different Schema .....              | 2-10 |

### 3 Installing Client Tools and Applications

|   |     |
|---|-----|
| <b>Oracle Spreadsheet Add-In for Predictive Analytics</b> ..... | 3-1 |
| Prerequisites .....   | 3-1 |
| Installing the Spreadsheet Add-In for Predictive Analysis ..... | 3-2 |
| Connecting Excel to Oracle Database .....                       | 3-2 |
| Creating an Oracle Net Service Name .....                       | 3-3 |
| <b>Oracle Data Miner</b> .....                                  | 3-4 |
| Installing Oracle Data Miner on Windows .....                   | 3-5 |
| Installing Oracle Data Miner on Linux .....                     | 3-5 |
| Prerequisite .....  | 3-5 |
| Installation Steps .....  | 3-5 |
| Connecting Oracle Data Miner to Oracle Database .....           | 3-5 |
| <b>Data Mining Application Programming Interfaces</b> .....     | 3-6 |
| PL/SQL Development Tools .....                                  | 3-7 |
| Java Development Tools .....                                    | 3-7 |

### 4 Installing and Using the Sample Programs

|   |     |
|---|-----|
| <b>Obtaining the Sample Programs</b> .....        | 4-1 |
| The PL/SQL Samples .....                          | 4-3 |
| The Java Samples .....                            | 4-4 |
| <b>Preparing to Run the Sample Programs</b> ..... | 4-5 |
| A Sample User Scenario .....                      | 4-5 |
| <b>Examining the Data</b> .....                   | 4-6 |
| Customer Data for Data Mining .....               | 4-6 |
| Market Basket Data for Association Rules .....    | 4-8 |

|  |      |
|--|------|
| Customer Data for Text Mining .....                                  | 4-8  |
| <b>Using the PL/SQL Samples</b> .....                                | 4-10 |
| Running the PL/SQL Programs .....                                    | 4-10 |
| PL/SQL Program Summaries .....                                       | 4-10 |
| SQL Scoring Functions .....  | 4-11 |
| <b>Using the Java Samples</b> .....                                  | 4-12 |
| Preparing to Run the Java Programs .....                             | 4-12 |
| Running the Java Programs .....                                      | 4-13 |
| Java Program Summaries .....   | 4-13 |
| <b>Using the Text Mining Samples</b> .....                           | 4-14 |
| Text Mining in PL/SQL .....  | 4-15 |
| Text Transformation Sample .....                                     | 4-15 |
| Text Transformation for the PL/SQL Sample Text Mining Programs ..... | 4-15 |
| The Sample Text Mining Models (PL/SQL) .....                         | 4-15 |
| Text Mining in Java .....  | 4-16 |
| Text Transformation for the Java Sample Text Mining Programs .....   | 4-16 |
| The Sample Text Mining Models (Java) .....                           | 4-16 |
| <b>Using the BLAST Sample</b> .....                                  | 4-16 |
| Preparing to Run the BLAST Demo .....                                | 4-17 |
| Running the BLAST Table Functions .....                              | 4-17 |



---

---

# Preface

The *Oracle Data Mining Administrator's Guide* explains how to install the various components of Oracle Data Mining.

The preface contains these topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)

## Audience

This guide is intended for a spectrum of users who need to install the software that supports Oracle Data Mining:

- Individual users who want to evaluate Oracle Data Mining
- Departmental DBAs who manage an instance of Oracle Database for Data Mining users
- Enterprise DBAs and IT professionals who need to support Data Mining users as part of administering Oracle Database for a larger user community

## Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

### Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

### Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

### TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

## Related Documents

For more information about Oracle Data Mining, see the following manuals in the Oracle Database 10g documentation set:

- *Oracle Data Mining Concepts*
- *Oracle Data Mining Java API Reference*
- *Oracle Data Mining Application Developer's Guide*
- *Oracle Database PL/SQL Packages and Types Reference* (Search for "Data Mining")
- *Oracle Database SQL Reference* (Search for "Data Mining")

The Oracle By Example (OBE) series provides hands-on, step-by-step instructions on how to implement various technology solutions to business problems. An OBE tutorial on Data Mining is available on the Oracle Technology Network at

<http://www.oracle.com/technology/obe/obe10gdb/bidw/odm/odm.htm>

## Conventions

The following text conventions are used in this document:

| Convention      | Meaning  |
|-----------------|--|
| <b>boldface</b> | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.         |
| <i>italic</i>   | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.                          |
| monospace       | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

---

---

# Installing Oracle Database for Data Mining

This chapter explains how to install Oracle Database to support Data Mining applications and development. It is directed primarily at users who want to install Oracle Database on a personal computer for their own use.

This chapter contains the following topics:

- [Who Needs to Install Oracle Database?](#)
- [Installation of Oracle Database 10g](#)
- [Tips for Linux Installations](#)
- [Additional Information for Enterprise Installations](#)
- [Verifying Installation of Data Mining Components](#)
- [Upgrading from Oracle Database 10g Release 1 \(10.1\)](#)

## Who Needs to Install Oracle Database?

If you want to use Oracle Data Mining, then you must have access to Oracle Database. You may be able to access an existing installation instead of installing Oracle Database yourself. Contact your Oracle database administrator and inquire whether you can access an instance of Oracle Database 10g Release 2 that is operating in 10.1 or later compatibility mode.

If you can access an existing installation, then ask your Oracle DBA to do the following:

1. Install the Sample Schemas, if they are not already installed.
2. Install the sample Data Mining programs, as described in "[Installing the Data Mining Sample Programs](#)" on page 1-7.
3. Create a user name for you, as described in "[Creating Oracle Database Users for Data Mining](#)" on page 2-4.
4. Provide you with the connection information.

Meanwhile, you can install a Data Mining client on your personal computer, as described in [Chapter 3](#).

If you do *not* have access to an existing installation, then follow the installation instructions provided here for installing Oracle Database.

## Installation of Oracle Database 10g

There are three installation disks for Oracle Database:

- **Database.** Installs Oracle Database 10g Release 2 Enterprise Edition
- **Companion.** Installs the Data Mining sample program files
- **Client.** Installs administrative tools for Oracle Database; for installation only on a local computer when Oracle Database is installed on a remote computer

Documentation is available on a separate CD or from the Oracle Technology Network.

This chapter provides step-by-step instructions for installing Oracle Database on a Microsoft Windows platform. If you use a different platform, you can still derive the preferred settings for Data Mining from these instructions. However, you should be careful to perform any pre- and post-installation steps specific to your platform, because they may be critical to a successful installation. Refer to the *Installation Guide* for your platform.

To install Oracle Database, you use Oracle Universal Installer. This installation program is nearly the same on all platforms, with differences that arise because of the differences in the operating systems.

### See Also:

- *Oracle Database Quick Installation Guide for Microsoft Windows (32-Bit)*
- *Oracle Database Installation Guide for Microsoft Windows (32-Bit)*
- *Oracle Database 2 Day DBA*

## Hardware Requirements

Verify that your computer meets the following requirements:

- Processor: 200 MHz minimum
- Available hard disk space:
  - System drive: 100 MB
  - Oracle home drive: 1.5 GB
  - Temp disk space: 100 MB
- RAM: 256 MB minimum, 512 MB recommended
- Virtual memory: double the amount of RAM
- Video adapter: 256 color

## Software Requirements

The following operating systems are among those supported for Oracle Database:

- Windows XP Professional
- Windows 2000 with service pack 1 or higher

## Preliminary Steps

If any Oracle products are already installed on your computer, you must disable them before installing Oracle Database. Take these steps:

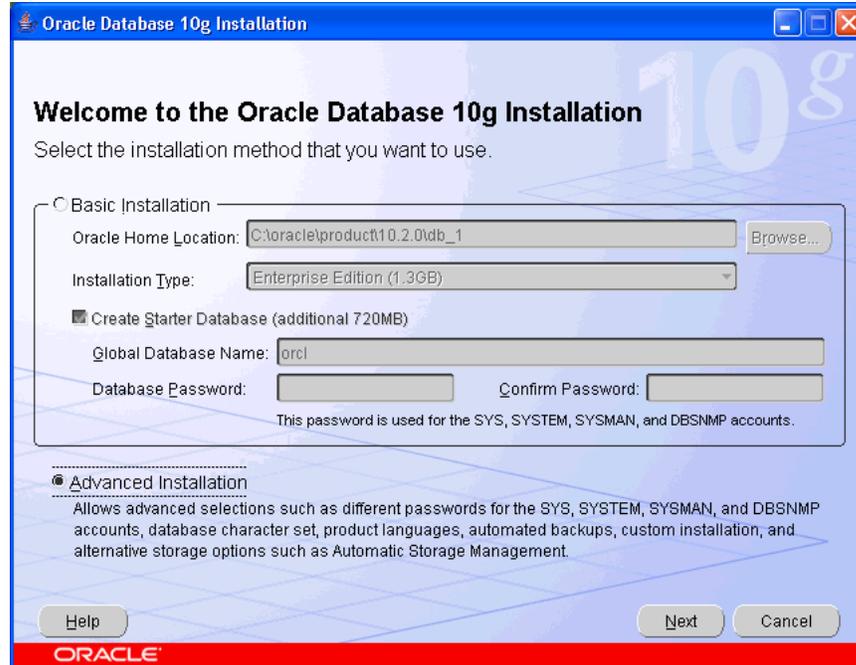
- Delete the ORACLE\_HOME environment variable:
  1. From the Control Panel, choose **System**.
  2. On the Advanced tab of the System Properties page, choose **Environment Variables**.
  3. In the System Variables, select ORACLE\_HOME, then click **Delete**.
- Stop all Oracle services.
  1. From the Control Panel, choose **Administrative Tools**, then **Services**.
  2. On the Services page, scroll down the list of names and locate those that begin with Oracle. Select those services and choose **Stop**.

If you are installing from a network drive, use Windows File Manager to map that drive to your computer.

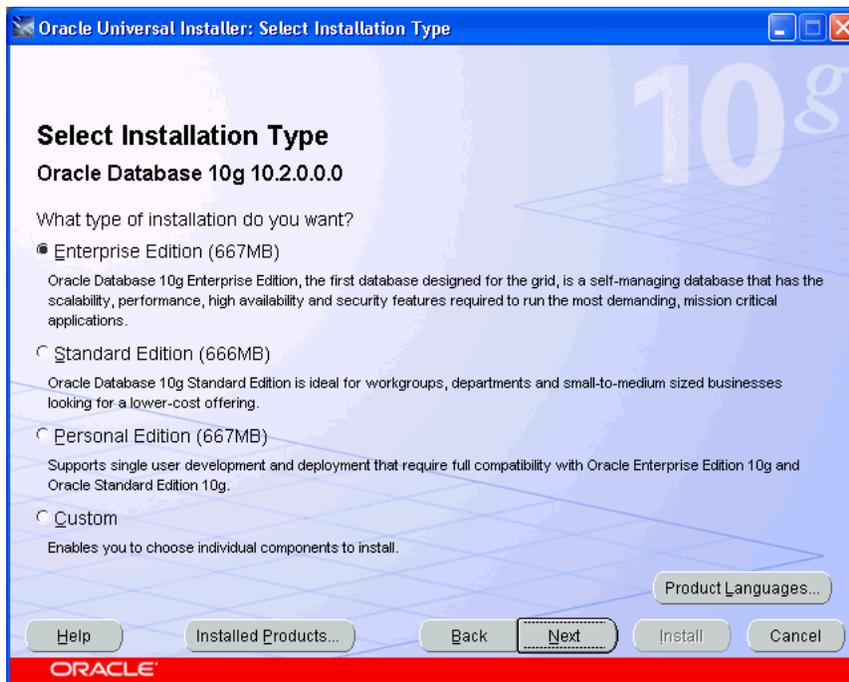
## Installing Oracle Database

To install Oracle Database on a Windows platform, take these steps:

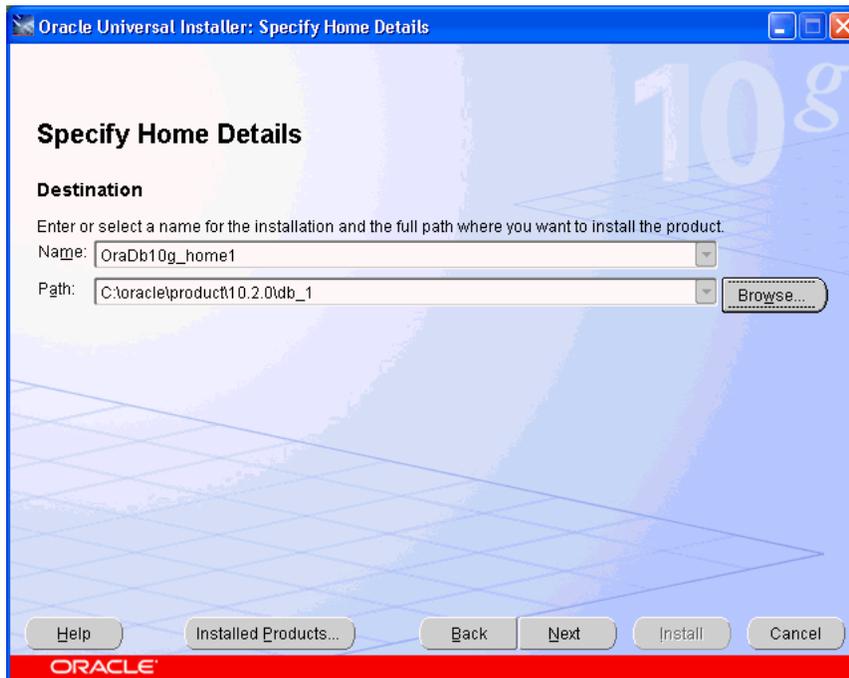
1. From Database (db) Disk 1 (of 1), run SETUP . EXE.  
Oracle Universal Installer opens and displays the Welcome page.
2. Select **Advanced Installation**. Click **Next** to advance to each page.



- On the Select Installation Type page, select **Enterprise Edition**.



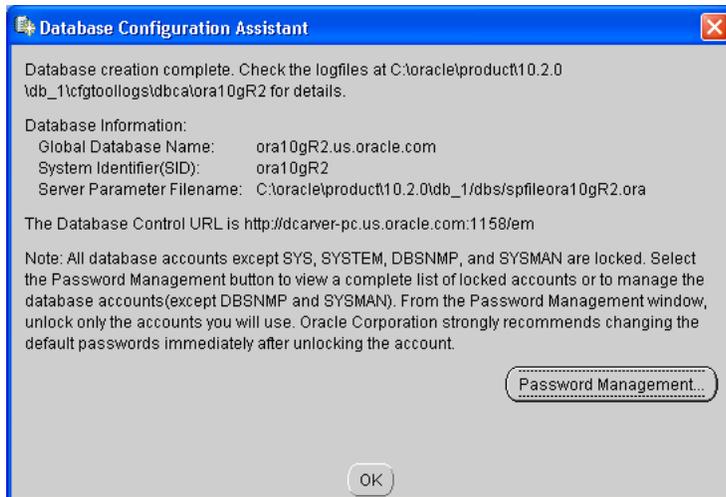
- On the Specify Home Details page, create a new Oracle home directory.



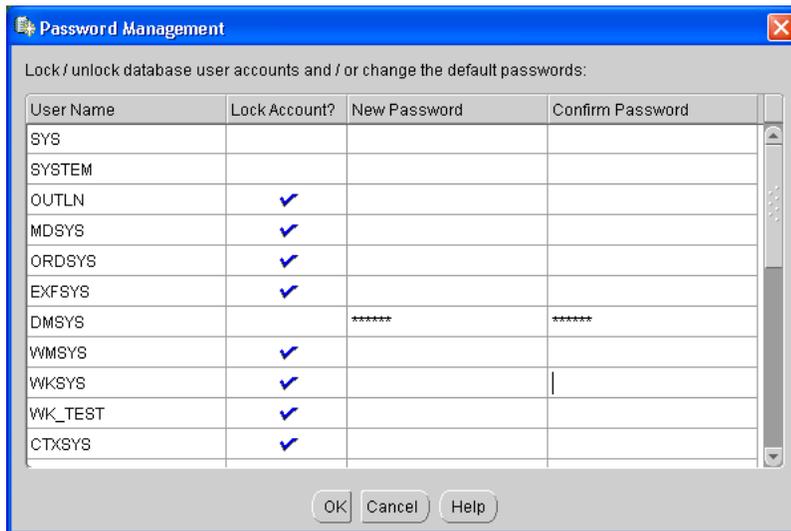
- On the Product-Specific Prerequisite Checks page, verify that all checks succeeded. If any checks failed, then you must correct the problem before proceeding.
- On the Select Configuration Option page, select **Create a database**.
- On the Select Database Configuration page, select **General Purpose**.
- Complete the Specify Database Configuration Options page as follows:

- Type in a global database name and SID.  
The SID is a name that you choose. If you are installing on a personal computer for your own use, then the SID and the global database name are the same; the global database name does not include the domain.
- Choose a database character set from the drop-down list. The Western European character set is a good choice for databases for the European Union and the Americas. UTF8 is a good choice for global databases.
- Select **Create database with sample schemas**.

9. Choose the options you want. The following steps identify the simplest choices.
  - a. On the Select Database Management Option page, select **Use Database Control for Database Management**.
  - b. On the Select Database File Storage Option page, select **File System**.
  - c. On the Specify Backup and Recovery Options page, select **Do not enable automated backups**.
  - d. On the Specify Database Schema Passwords page, select **Use the same password for all the accounts**. Then enter a password of your choice in the **Enter Password** and **Confirm Password** fields.
10. On the Summary page, review your previous choices, then click **Install**.
11. On the Database Configuration Assistant page, click **Password Management**.  
**Tip:** Print a screen shot of this page or write down the information it contains for future reference.

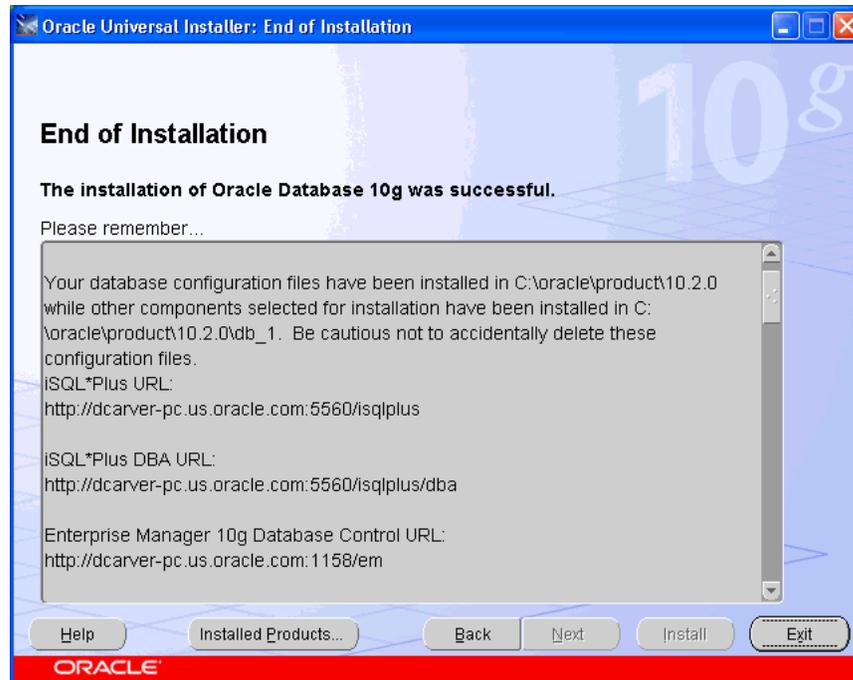


12. On the Password Management page, unlock the DMSYS, SCOTT, and SH accounts and enter new passwords for them.



13. On the End of Installation page, confirm that the installation was successful.

**Tip:** Print a screen capture of this page or write down the information it contains for future reference.



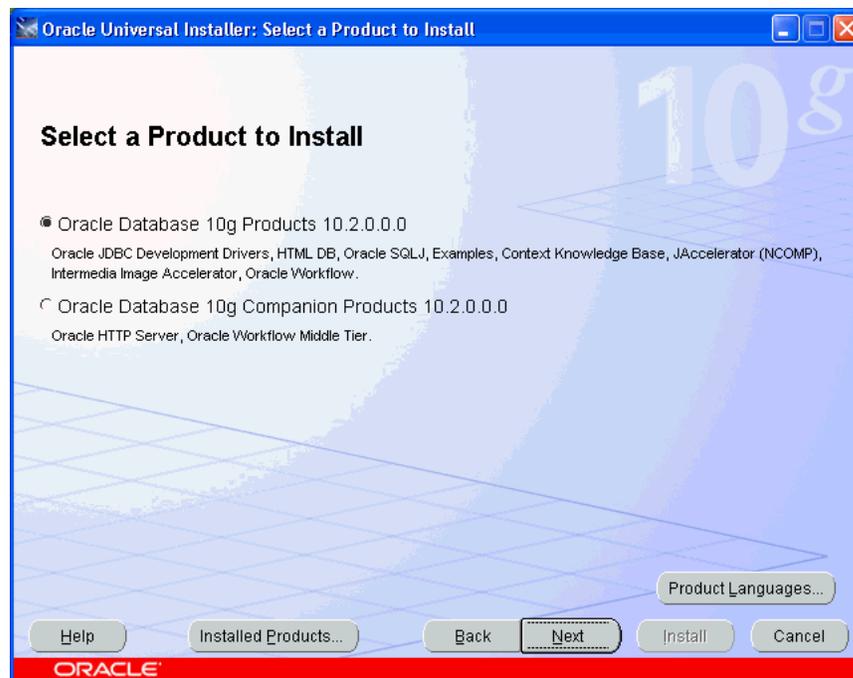
## Installing the Data Mining Sample Programs

To install the Data Mining Sample Program Files on a Windows platform, take these steps:

1. From Companion Disk 1 (of 1), run `SETUP . EXE`.

Oracle Universal Installer opens and displays the Welcome page. Click **Next** to advance to each page.

2. On the Select a Product to Install page, select **Oracle Database 10g Products 10.2.0.0.0**.



3. On the Specify Home Details page, select the Oracle home in which you installed Oracle Database 10g Release 2 in the previous procedure. Do not rely on the default setting to be correct.
4. On the Product-Specific Prerequisite Checks page, verify that all checks succeeded. If any checks failed, then you must correct the problem before proceeding.
5. On the Summary page, review your previous choices, then click **Install**.
6. On the End of Installation page, confirm that the installation was successful.

## Tips for Linux Installations

The step-by-step instructions in this guide are for installations on a Windows platform. The following are some tips to help you with the differences between the Linux and Windows platforms.

- Follow the instructions in the installation guide for your platform. Be sure to set system variables and perform any other pre-installation tasks.
- During the installation, you must run some SQL scripts as `root`, so make sure that you have root access.
- To run Oracle Universal Installer, type `runInstaller` from the installation disk or directory.
- Create a script for setting the appropriate environment variables, or include the settings in the initialization script for your operating system login (such as `.profile` or `.cshrc`). Perform any other post-installation tasks.

The following is a sample script for setting environment variables:

```
setenv ORACLE_ROOT /dat1/10gR2
setenv ORACLE_BASE ${ORACLE_ROOT}
setenv ORACLE_HOME ${ORACLE_BASE}/oracle/product/10.2.0/db_1
setenv ORACLE_PORT 1521
setenv ORACLE_SID re110g
setenv TNS_ADMIN ${ORACLE_HOME}/network/admin
setenv PATH ${ORACLE_HOME}/bin:${ORACLE_HOME}:%PATH
```

### See Also:

- *Oracle Database Quick Installation Guide for Linux x86* for a basic installation
- *Oracle Database Installation Guide for Linux x86* for a custom installation, upgrade, or other variation

## Additional Information for Enterprise Installations

The information in this section is for DBAs who plan to install Oracle Database on Real Application Clusters (RAC) or deploy Data Mining models in a different database than the one in which they were developed.

### Creating Data Mining Tablespaces on Real Application Clusters

When Oracle Database is installed on RAC, the Data Mining tables should be stored in a tablespace with a raw device partition. You can define this tablespace during installation of Oracle Database by choosing **Raw Devices** on the Specify Database File Storage Option page. Pre-installation steps are required for this storage option. Refer to the *Installation Guide* for your platform for more information.

## Installing the Data Mining Scoring Engine

You only need to install the Data Mining Scoring Engine on computers where you will apply models that were built on another computer system. The Scoring Engine is an Enterprise Edition option that can be installed only during a custom installation. You can include the Data Mining Scoring Engine in a new installation, or you can add it to an existing installation at a later time.

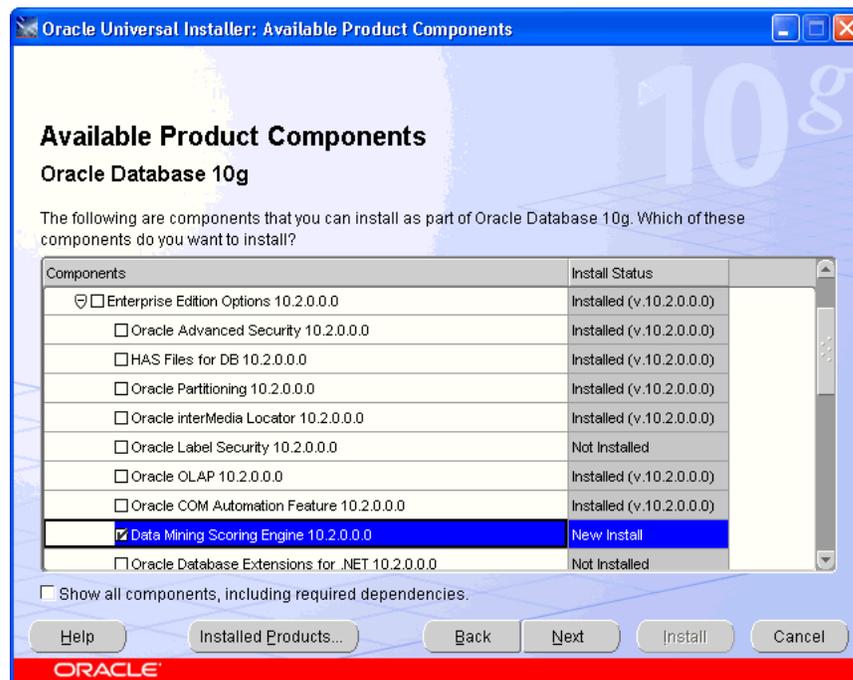
---

**Note:** The Data Mining Scoring Engine and the Data Mining option cannot be installed on the same computer. Installation of one option will de-install the other option. No other database options are affected by installation of the Data Mining Scoring Engine.

---

To add the Data Mining Scoring Engine to an existing installation of Oracle Database, take these steps:

1. Open Oracle Universal Installer as described in "[Installation of Oracle Database 10g](#)" on page 1-2.
2. On the Select Installation Type page, select **Custom**.
3. On the Specify Home Details page, select the Oracle home in which you previously installed Oracle Database 10g Release 2. Do not rely on the default setting to be correct.
4. On the Available Product Components page, select **Data Mining Scoring Engine**.



## Verifying Installation of Data Mining Components

If Data Mining is installed, then the following SQL query returns TRUE:

```
SELECT VALUE FROM V$OPTION WHERE PARAMETER = 'Data Mining';
```

If the Data Mining Scoring Engine is installed, then the next SQL query returns TRUE:

```
SELECT VALUE FROM V$OPTION WHERE PARAMETER = 'Data Mining Scoring Engine';
```

For more detailed information, query DBA\_REGISTRY with a command such as this one:

```
SQL> SELECT COMP_ID, VERSION, STATUS FROM DBA_REGISTRY WHERE COMP_ID = 'ODM';
```

| COMP_ID | VERSION    | STATUS |
|---------|------------|--------|
| ODM     | 10.2.0.0.0 | VALID  |

## Upgrading from Oracle Database 10g Release 1 (10.1)

If Oracle Database 10g Release 1 is installed on your computer, then install Oracle Database 10g Release 2 into a separate Oracle home. A new version of the Data Mining option is installed along with the other components of Oracle Database.

During installation of release 10.2, Oracle Universal Installer will detect 10.1 databases and will prompt you to upgrade them. For upgrading, the Installer will open the Database Upgrade Assistant (DBUA). It automatically upgrades these Data Mining components:

- Databases
- DMSYS repository schema
- PL/SQL data models in user schemas

**See Also:** *Oracle Database Upgrade Guide* for complete upgrade instructions

## Post-Upgrade Steps for Data Mining

The data mining objects in these schemas require additional upgrade steps after upgrading the database:

- DMSYS Repository Schema. The `odmu101s.sql` script removes redundant release 10.1 objects.
- User Schemas. The `odmu101a.sql` script re-creates the views of the Sales History sample schema for users who wish to run the sample Data Mining programs described in [Chapter 4](#).

To upgrade the DMSYS repository, follow these steps:

1. Open SQL\*Plus and connect to the database as the SYS user or another privileged account.
2. Verify that the compatibility mode is set to 10.2.x.x (such as 10.2.0.0).

```
SELECT name, value, description FROM v$parameter
WHERE name='compatible';
```

If the compatibility model is incorrect, then edit the `init.ora` file for the database.

3. Run the `odmu101s.sql` script.

Issue this command on Windows platforms:

```
%ORACLE_HOME%\rdbms\admin\odmu101s.sql
```

Or this command on Linux platforms:

```
$ORACLE_HOME/rdbms/admin/odmu101s.sql
```

To upgrade the sample Sales History views in the schemas of Data Mining users, take these steps for each schema:

1. Backup the schema using the `expdp` command in Oracle Data Pump.
2. Open SQL\*Plus and connect to the database as the `SYS` user or another privileged account.
3. Run the `odmu101a.sql` script, passing it two parameters: `user_name` and `password`.

Issue this command on Windows platforms:

```
%ORACLE_HOME%\rdbms\admin\odmu101a.sql user_name password
```

Or this command on Linux platforms:

```
$ORACLE_HOME/rdbms/admin/odmu101a.sql user_name password
```

## Upgrading PL/SQL Models

All data mining PL/SQL models residing in user schemas in a release 10.1 database can be upgraded as part of the database upgrade. The upgrade of models is seamlessly integrated into the database upgrade process.

You can also export models from a release 10.1 database and import them into a release 10.2 database. The import process upgrades the models to release 10.2. Use the `expdp` and `impdp` command utilities in Oracle Data Pump. The Data Mining option or the Data Mining Scoring Engine option must be installed in the target database.

---



---

**Note:** Java models cannot be upgraded. They must be re-created in Oracle Database 10.2.

---



---

The Database Upgrade Assistant invokes procedures in the Data Mining `DMP_SYS` public package for upgrading and downgrading PL/SQL models between release 10.1 and 10.2.

The `UPGRADE_MODELS` procedure upgrades the Data Mining models from release 10.1 to 10.2, and has this syntax:

```
UPGRADE_MODELS (
    to_version    IN    VARCHAR2);
```

The `DOWNGRADE_MODELS` procedure downgrades models from release 10.2 to 10.1, and has this syntax:

```
DOWNGRADE_MODELS (
    from_version  IN    VARCHAR2);
```

These procedures are integrated with the Data Mining upgrade and downgrade processes and run as `SYS`. You do not need to run them manually.

## Downgrading Oracle Data Mining

You can downgrade a database from release 10.2 to release 10.1 if the compatibility setting is set to 10.1.x.x. Otherwise, the database will include incompatibilities that will prevent it from being downgraded. The Data Mining option and user-owned PL/SQL models will be downgraded as part of database downgrade.

**See Also:** *Oracle Database Upgrade Guide* for complete downgrade instructions

---

---

# Administering Oracle Database for Data Mining

Because Oracle Data Mining is completely integrated with Oracle Database, you use the same tools for administering Data Mining as you would a database for any other purpose.

In this chapter, you will learn about some post-installation administrative tasks, such as creating a sample schema that can be used for data mining, and creating database users with sufficient privileges to mine the data.

This chapter contains the following topics:

- [Introduction to Oracle Administration Tools](#)
- [Creating a Sample Schema for Data Mining](#)
- [Creating Oracle Database Users for Data Mining](#)
- [Exporting and Importing Data Mining Models](#)

## Introduction to Oracle Administration Tools

You can administer Oracle Database locally or from a remote computer with network access.

### Local Administration on Microsoft Windows

Several tools for administrators and application developers are installed along with Oracle Database. For Microsoft Windows platforms, the Program menu contains an Oracle home program group with links to the tools.

Following are descriptions of a few of the basic administrative tools.

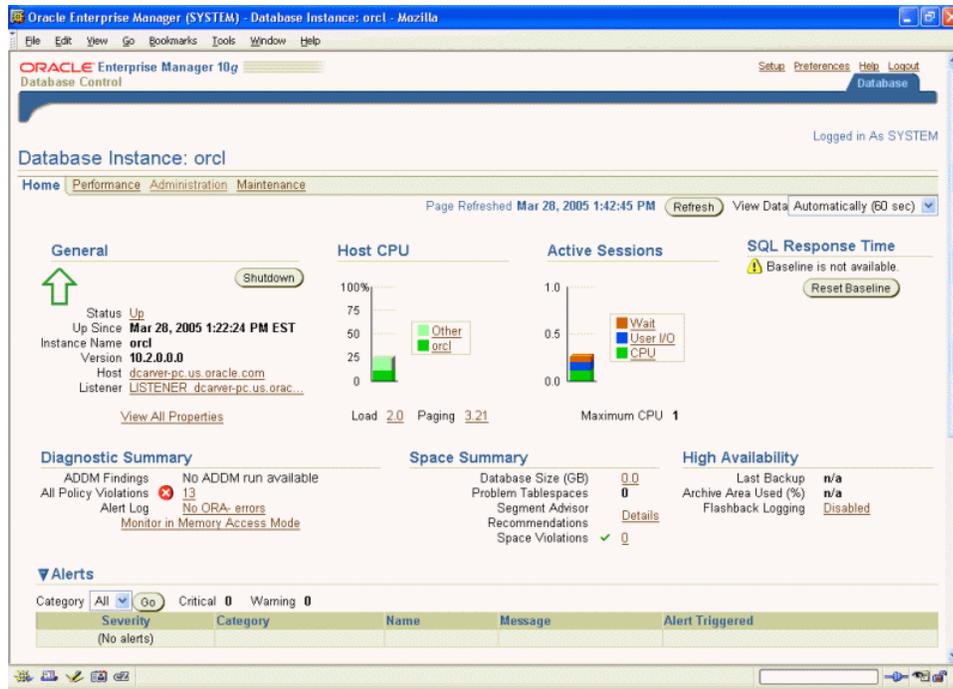
#### Oracle Enterprise Manager Database Control

Database Control provides a Web-based graphical interface for managing all aspects of Oracle Database.

To open Database Control, click **Start > All Programs > Oracle - *oracle\_home* > Database Control - *database\_instance***.

You can also open Database Control from the URL provided during installation.

The following figure shows the Database Control home page.



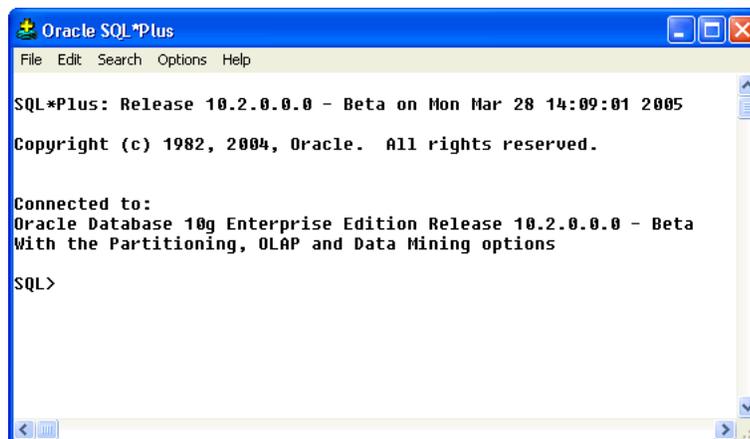
## SQL\*Plus

SQL\*Plus is a command-line interface for the SQL language. You can perform all Oracle administrative tasks using SQL.

To open SQL\*Plus, click **Start > All Programs > Oracle - oracle\_home > Application Development > SQL Plus**.

You will be prompted for your user name and password. You must supply a host string only when connecting to a remote computer. The host string takes the form *host\_name:port:SID*, such as *myhost:1521:orcl*.

The following figure shows the SQL Plus window.



## Database Configuration Assistant

Database Configuration Assistant provides a graphical user interface for creating, configuring, and deleting database instances. A single installation of Oracle Database can support numerous individual database instances. You can use Database

Configuration Assistant to install the sample schemas if you did not install them with the database.

To open Database Configuration Assistant, click **Start > All Programs > Oracle - oracle\_home > Configuration and Migration Tools > Database Configuration Assistant**.

### Oracle Universal Installer

You can use Oracle Universal Installer to list the Oracle products on your computer or to deinstall them.

To open Oracle Universal Installer, click **Start > All Programs > Oracle - oracle\_home > Oracle Installation Products > Universal Installer**.

You must shut down all databases and supporting services before deinstalling Oracle Database. Refer to the installation guide for your platform for more information.

### Oracle Services

The Oracle Database installation creates several services. The following table describes some of them.

| Service Name               | Description  | Usage   |
|----------------------------|--|---|
| OracleServiceSID           | Oracle Database                                    | Enables you to start and stop Oracle Database from the Service window.        |
| OracleHome_NameTNSListener | Oracle Database listener                           | Enables you to open a connection with Oracle Database from a remote computer. |
| OracleHome_NameiSQL*Plus   | iSQL*Plus application server                       | Enables you to open iSQL*Plus from a browser.                                 |
| OracleDBConsoleSID         | Oracle Enterprise Manager Database Control console | Enables you to open Database Control from a browser.                          |

To manage them, open Administrative Tools in the Windows Control Panel and choose **Services**.

## Local Administration on Linux

The same tools that are installed locally on a Windows platform are also installed on Linux. You can run the local administrative tools from the shell command line. They are located in `$ORACLE_HOME/bin`. These are a few of the tools:

- To open SQL\*Plus, type `sqlplus`.
- To open Database Configuration Assistant, type `dbca`.
- To open Enterprise Manager Database Control, open a browser and type the URL provided during installation.
- To open Oracle Universal Installer, type `$ORACLE_HOME/oui/bin/runInstaller`.
- To start and stop the various Oracle processes, use these commands:
  - `lsnrctl`: Oracle Database listener

- isqlplusctl: iSQL\*Plus application server
- emctl: Oracle Enterprise Manager Database Control console

For descriptions of these tools, refer to "[Local Administration on Microsoft Windows](#)".

## Remote Administration

You can open these tools in any browser by typing the URLs listed during installation on the End of Installation page:

- iSQL\*Plus is a version of SQL\*Plus that runs in a browser.
- Enterprise Manager Database Control is the same thin-client application that you access locally.

If you prefer, you can install on a client computer all of the same tools that are installed on the host computer with Oracle Database. They are available on Client Disk 1 (of 1).

## Creating Oracle Database Users for Data Mining

Anyone who wants to use Oracle Database must have a user name and password. Oracle Data Mining requires a small number of database permissions, plus `SELECT` access to the tables containing data for analysis.

### Creating a Demo User for Data Mining

If you plan to use the Data Mining demo programs, then take these steps:

1. Open SQL\*Plus and log in as the `SYSTEM` user.
2. Create a user name. The following command creates a user named `dmuser` with the password `change_now`, and provides default access to two tablespaces shared by several other sample schemas:

```
CREATE USER dmuser IDENTIFIED BY change_now
        DEFAULT TABLESPACE users
        TEMPORARY TABLESPACE temp
        QUOTA UNLIMITED on users;
```

3. Run the `dmshgrants` SQL script, which assigns all of the necessary permissions. The following command passes two arguments to the script: The password for the `SH` user (which is also `SH` in this example) and the user name getting the access rights (`DMUSER`).

```
@%ORACLE_HOME%\rdbms\demo\dmshgrants sh dmuser
```

On Linux, this is the equivalent command:

```
@$ORACLE_HOME/rdbms/demo/dmshgrants sh dmuser
```

### Creating Analytic Users for Data Mining

An analyst who wants to mine real data needs to have the appropriate resources. These resources include:

- Personal tablespaces (permanent and temporary)
- Access rights to the data

The examples in this guide show how to allocate these resources using SQL commands, which you can cut and paste into SQL\*Plus. However, you can use Enterprise Manager if you prefer using a graphical interface.

### Creating Default Tablespaces

All users require a permanent tablespace and a temporary tablespace in which to do their work. Performance may start to degrade if multiple users are sharing the same tablespace while mining large data sets. You can improve performance by creating individual tablespaces for each user.

The following SQL command creates a new permanent tablespace.

```
CREATE TABLESPACE "ODMPERM" DATAFILE
  'C:\ORACLE\PRODUCT\10.2.0\ORADATA\ORCL\odm1.dbf'
  SIZE 20M REUSE AUTOEXTEND ON NEXT 20M;
```

The next SQL command creates a new temporary tablespace.

```
CREATE TEMPORARY TABLESPACE "ODMTEMP" TEMPFILE
  'C:\ORACLE\PRODUCT\10.2.0\ORADATA\ORCL\odmtemp.tmp'
  SIZE 20M REUSE AUTOEXTEND ON NEXT 20M;
```

### Creating Database Users for Data Mining

To create a user for Data Mining, you use the same SQL command as the one shown in ["Creating a Demo User for Data Mining"](#) on page 4, except that you may identify personal tablespaces. The following example shows how to change the default tablespaces for an existing user.

```
ALTER USER dmuser DEFAULT TABLESPACE odmperm
  DEFAULT TEMPORARY TABLESPACE odmtemp
  QUOTA UNLIMITED ON odmperm;
```

## Granting Access Rights

You can grant access rights to users by running the `dmsgrants` PL/SQL script, as described in ["Creating a Demo User for Data Mining"](#), or using the graphical interface provided by Enterprise Manager Database Control, or issuing SQL commands.

### What the `dmsgrants` Script Does

The `dmsgrants` script grants the following database permissions. If you do not have access to the script, you can set permissions using the SQL `GRANT` command or Enterprise Manager Database Control.

```
CREATE PROCEDURE
CREATE SESSION
CREATE TABLE
CREATE SEQUENCE
CREATE VIEW
CREATE JOB
CREATE TYPE
CREATE SYNONYM
```

The `dmsgrants` script grants `SELECT` rights on these tables in the `SH` schema:

```
COUNTRIES
CUSTOMERS
PRODUCTS
```

```
SUPPLEMENTARY_DEMOGRAPHICS  
SALES
```

For text mining, `dmshgrants` grants access rights to an Oracle Text package:

```
EXECUTE ON ctxsys.ctx_ddl
```

### What the `dmshgrants` Script Does Not Do

The `dmshgrants` script does not grant access rights to any data other than the Sales History sample schema. Unless a user owns the data being analyzed, you must grant access rights to that data using a SQL command like this one:

```
GRANT SELECT ON owner.tablename TO user
```

For example, the following SQL command grants `SELECT` access to the `EMPLOYEES` table in the sample `HR` schema to `DMUSER`.

```
GRANT SELECT ON hr.employees TO dmuser
```

Users who want to export and import Data Mining models need additional access rights, as described in "[Exporting and Importing Data Mining Models](#)" on page 2-6.

## About the `DMSYS` Schema

Information about all models created in a database is stored in tables owned by the `DMSYS` user. During a typical installation, the `DMSYS` user has `SYSAUX` defined as its default tablespace.

Do not delete, truncate, or modify the tables in the `DMSYS` schema. They support the data mining activities of all users in the database.

## Creating a Sample Schema for Data Mining

The sample programs for Data Mining reference the `SH` schema and some additional tables and views. You can create the new tables and views in the `DMUSER` schema so that all users can share them, or individual users can create the tables and views in their own schemas.

To create the data sets used by the Data Mining sample programs, take these steps:

1. Open SQL\*Plus and connect as the user who will own the new tables and views.

This user must have all of the privileges granted by the `dmshgrants` script, as described in "[Creating Oracle Database Users for Data Mining](#)". For example, you might connect as `dmuser`.

2. Run the `dmsh` SQL script, using this command on Windows:

```
@%ORACLE_HOME%\rdms\demo\dmsh
```

On Linux, this is the equivalent command:

```
@$ORACLE_HOME/rdms/demo/dmsh
```

## Exporting and Importing Data Mining Models

You can export data mining models to flat files to back up work in progress or to move models to a different instance of Oracle Database Enterprise Edition (such as from a development database to a production database). All methods for exporting and importing models are based in Oracle Data Pump technology.

Oracle Data Pump consists of two command-line clients and two PL/SQL APIs. The command-line clients, `expdp` and `impdp`, provide an easy-to-use interface to the Data Pump export and import utilities. The Data Mining APIs also use the Data Pump export and import utilities.

You can export and import models at different levels, depending on your access rights in the database:

- **Database.** When a DBA exports a full database using `expdp`, an Oracle Data Pump utility, all data mining models in the database are exported. Another Data Pump utility, `impdp`, imports all the models with the other objects in the database.
- **Schema.** When a DBA or an individual user exports a schema using `expdp`, all the data mining models in the schema are exported. Likewise, `impdp` imports all the models with the other objects in the schema.
- **Models Only.** The Data Mining APIs contain utilities for exporting and importing either all Data Mining models in a schema or models that match specific criteria.

The Data Pump export utility writes the tables and metadata that constitute a model to a dump file set, which consists of one or more files. The Data Pump import utility retrieves the tables and metadata from the dump file and restores them to the target database. Because the `expdp` and `impdp` clients and the Data Mining APIs use the Data Pump export and import utilities, you can use the APIs to extract individual models from a dump file of a schema or database.

Note that the older `exp` and `imp` database utilities do not export or import data mining models.

#### See Also:

- *Oracle Database Utilities* for a complete discussion of Oracle Data Pump and the `expdp` and `impdp` utilities
- *Oracle Database PL/SQL Packages and Types Reference* for detailed information about the Data Mining APIs

## Prerequisites

To export and import Data Mining models, you must have read and write access to a directory object, and you may need additional database permissions.

### Directory Objects

A directory object is a logical name in the database for a physical directory on the host computer. Without read and write access to a directory object, you cannot access the host computer file system from within Oracle Database.

You must have the `CREATE ANY DIRECTORY` privilege to create directory objects.

The following SQL command creates, or re-creates if it already exists, a directory object named `DMTEST`. The file system directory (in this example, `C:\ORACLE\PRODUCT\10.2.0\DMINING`) must already exist and have shared read/write access rights granted by the operating system.

```
CREATE OR REPLACE DIRECTORY dmtest AS 'c:\oracle\product\10.2.0\dmining';
```

This SQL command gives user `DMUSER` both read and write access to `DMTEST`.

```
GRANT ALL ON DIRECTORY dmtest TO dmuser;
```

For more information about creating database directories, refer to the `CREATE DIRECTORY` and `GRANT` commands in the *Oracle Database SQL Reference*.

### Additional Database Privileges

You may need special privileges in the database to take full advantage of all Data Pump features, such as importing models and other objects into a different schema. These privileges are granted by the `EXP_FULL_DATABASE` and `IMP_FULL_DATABASE` roles.

You do not need these roles to export models from your own schema. To import models, you must have the same database roles or be as privileged as the user who created the dump file set. Otherwise, you need the `IMP_FULL_DATABASE` role.

Privileged users (such as `SYS` or a user with the `DBA` role) have sufficient access rights and do not need these additional roles.

The following SQL commands grant these roles to `DMUSER`:

```
GRANT EXP_FULL_DATABASE TO dmuser;  
GRANT IMP_FULL_DATABASE TO dmuser;
```

## PL/SQL APIs for Exporting and Importing Models

The `DBMS_DATA_MINING` PL/SQL package contains these two procedures:

- `EXPORT_MODEL`
- `IMPORT_MODEL`

For more information about these procedures, refer to the *Oracle Database PL/SQL Packages and Types Reference*.

## Java APIs for Exporting and Importing Models

Oracle Database implements the industry-standard Java Data Mining (JDM) API Specification, which includes these two interfaces:

- `javax.datamining.task.ExportTask`
- `javax.datamining.task.ImportTask`

For more information about the standard JDM API, refer to the Java Help for the JSR-73 Specification, which is available on the Oracle Technology Network at

<http://www.oracle.com/technology/products/bi/odm/JSR-73/index.html>

## Tables Created By Exporting and Importing Models

Two tables are created in the user's schema by the Data Mining export and import utilities:

- `DM$P_MODEL_EXPIMP_TEMP`. Used for internal purposes during export and import, and provides a job history.
- `DM$P_MODEL_TABKEY_TEMP`. Used only for internal purposes during export and import.

Do not alter these tables. However, you may drop them when no export or import job is running. The utilities will re-create them for the next job.

## Example: Exporting and Importing Models

This example creates a dump file with three models and imports the models from the dump file.

### Exporting Models from the DMUSER Schema

The following command exports all models from DMUSER, who is currently connected to the database in SQL\*Plus.

```
SQL> EXECUTE DBMS_DATA_MINING.EXPORT_MODEL('allmodels.dmp', 'DMTEST');
```

PL/SQL procedure successfully completed.

An export or import creates a log file in the same directory as the dump file. Error messages are returned to the current output device (such as the screen), and the log file may provide additional information.

This command was successful and creates two files in the DMTEST directory:

- A dump file named `allmodels01.dmp` (note the 2-digit suffix added to the name)
- A log file with a default name of `DMUSER_exp_4589.log`

For detailed information about the default names of files, see the `DBMS_DATA_MINING` package in the *Oracle Database PL/SQL Packages and Types Reference*.

You can view the log file using a system command or editor. You must know the path of the physical directory in order to locate the file.

`DMUSER_exp_4589.log` lists the three Data Mining models that were in the schema, plus additional objects as shown here:

```
Starting "DMUSER"."DMUSER_exp_45":  DM_EXPIMP_JOB_ID=45
Estimate in progress using BLOCKS method...
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
Total estimation using BLOCKS method: 1.062 MB
>>> . . exported Data Mining Model "DMUSER"."ABN_CLAS_SAMPLE"
>>> . . exported Data Mining Model "DMUSER"."ASSOCIATION_RULES_SAMPLE"
>>> . . exported Data Mining Model "DMUSER"."NAIVE_BAYES_SAMPLE"
Processing object type TABLE_EXPORT/TABLE/PROCACT_INSTANCE
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processing object type TABLE_EXPORT/TABLE/INDEX/INDEX
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/INDEX/FUNCTIONAL_AND_BITMAP/INDEX
Processing object type
TABLE_EXPORT/TABLE/INDEX/STATISTICS/FUNCTIONAL_AND_BITMAP/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
. . exported "DMUSER"."DM$P0ASSOCIATION_RULES_SAMPLE"      7.640 KB      15 rows
. . exported "DMUSER"."DM$P0NAIVE_BAYES_SAMPLE"           18.35 KB     219 rows
. . exported "DMUSER"."DM$P1ABN_CLAS_SAMPLE"              6.945 KB      2 rows
. . exported "DMUSER"."DM$P1NAIVE_BAYES_SAMPLE"           5.929 KB      2 rows
. . exported "DMUSER"."DM$P2ASSOCIATION_RULES_SAMPLE"     6.210 KB     11 rows
. . exported "DMUSER"."DM$P3ASSOCIATION_RULES_SAMPLE"     6.179 KB     18 rows
. . exported "DMUSER"."DM$P4ASSOCIATION_RULES_SAMPLE"     5.492 KB     26 rows
. . exported "DMUSER"."DM$P5ABN_CLAS_SAMPLE"              5.304 KB      2 rows
. . exported "DMUSER"."DM$P5NAIVE_BAYES_SAMPLE"           5.984 KB     27 rows
. . exported "DMUSER"."DM$P6ABN_CLAS_SAMPLE"             16.47 KB     34 rows
. . exported "DMUSER"."DM$P7ABN_CLAS_SAMPLE"              7.007 KB      5 rows
. . exported "DMUSER"."DM$P8ABN_CLAS_SAMPLE"              5.414 KB      5 rows
```

```

. . exported "DMUSER"."DM$P8ASSOCIATION_RULES_SAMPLE"      5.335 KB      3 rows
. . exported "DMUSER"."DM$P8NAIVE_BAYES_SAMPLE"           5.359 KB      3 rows
. . exported "DMUSER"."DM$PEABN_CLAS_SAMPLE"              9.093 KB     116 rows
. . exported "DMUSER"."DM$PENAIVE_BAYES_SAMPLE"          8.742 KB     116 rows
. . exported "DMUSER"."DM$P_MODEL_EXPIMP_TEMP"           6.273 KB      10 rows
. . exported "DMUSER"."DM$PEASSOCIATION_RULES_SAMPLE"      0 KB          0 rows
Master table "DMUSER"."DMUSER_exp_45" successfully loaded/unloaded
*****
Dump file set for DMUSER.DMUSER_exp_45 is:
  /dat2/10gR2/oracle/product/10.2.0/db_1/dmtest/allmodels01.dmp
Job "DMUSER"."DMUSER_exp_45" successfully completed at 08:40:08

```

### Importing Models Into the Same Schema

DMUSER can restore these models from the dump file at a later date if, for whatever reason, he or she wants to revert to this version of the models. Note that an import will not overwrite an existing model with the same name unless the model is incomplete or corrupted.

The following command restores all models from the dump file to the DMUSER schema:

```
SQL> EXECUTE DBMS_DATA_MINING.IMPORT_MODEL('allmodels01.dmp', 'DMTEST');
```

### Importing Models Into a Different Schema

A user with the necessary privileges can load the models from a dump file into a different schema. In the next example, the SYSTEM user issues the following command, which loads the three models into the SCOTT schema:

```
SQL> EXECUTE DBMS_DATA_MINING.IMPORT_MODEL('allmodels01.dmp', 'DMTEST', null,
null, null, 'toscott', 'DMUSER:SCOTT');
```

This import command specifies `toscott.log` as the name of the log file; the `.log` extension is added automatically to the name. The log file shows the names of the imported models and supporting metadata.

```

Master table "SYSTEM"."toscott" successfully loaded/unloaded
Starting "SYSTEM"."toscott":  DM_EXPIMP_JOB_ID=51|DM_SELECT_IMPORT
Processing object type TABLE_EXPORT/TABLE/PROCACT_INSTANCE
>>> . . imported Data Mining Model "SCOTT"."ABN_CLAS_SAMPLE"
>>> . . imported Data Mining Model "SCOTT"."ASSOCIATION_RULES_SAMPLE"
>>> . . imported Data Mining Model "SCOTT"."NAIVE_BAYES_SAMPLE"
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
Processing object type TABLE_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processing object type TABLE_EXPORT/TABLE/INDEX/INDEX
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Job "SYSTEM"."toscott" completed with 1 error(s) at 09:08:12

```

---

---

## Installing Client Tools and Applications

Oracle provides several tools that support data mining activities by both analysts and application developers. Users must have the database and object permissions described in [Chapter 2](#) to use these tools with Oracle Database.

This chapter contains the following topics:

- [Oracle Spreadsheet Add-In for Predictive Analytics](#)
- [Oracle Data Miner](#)
- [Data Mining Application Programming Interfaces](#)

### Oracle Spreadsheet Add-In for Predictive Analytics

The Oracle Spreadsheet Add-In for Predictive Analytics enables Microsoft Excel users to mine their Oracle Database or Excel data using simple "one click" Predict and Explain predictive analytics features. Predictive analytics provide automated methodologies that simplify data mining. Using Predictive Analytics, many more users can harness the power of Oracle Data Mining without possessing the knowledge of advanced data analysts.

### Prerequisites

The following components must be installed on your PC:

- Microsoft Excel 2000, 2002, or 2003
- Oracle Database 10g or Oracle Client 10g with these options:
  - Oracle Objects for OLE
  - Net Configuration Assistant

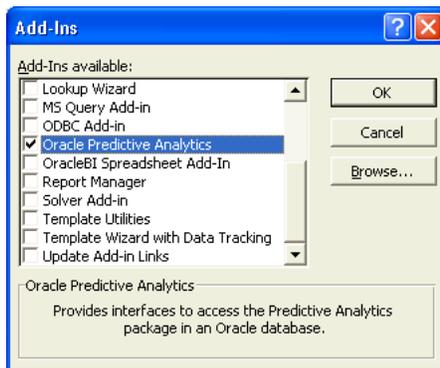
These options are included with Oracle Database Enterprise Edition and Oracle Client Administrator. You can also add them with a custom installation. Oracle Objects for OLE is a component of Oracle Windows Interfaces.

To check whether you have these components, check the Windows Start menu in the folder for your Oracle home installation. Look for Oracle Objects for OLE in the Application Development folder, and look for Net Configuration Assistant in the Configuration and Migration Tools folder.

## Installing the Spreadsheet Add-In for Predictive Analysis

To install the Spreadsheet Add-In on your personal computer, take these steps:

1. Download the Spreadsheet Add-In for Predictive Analysis from the Oracle Web site at <http://www.oracle.com/technology/products/bi/odm/index.html>.
2. Open the ZIP file and extract the file named `Predictive_Analytics.xla` to the Microsoft Office Library directory. The library has a path such as this one:  
C:\Program Files\Microsoft Office\Office\Library
3. Open Excel and click **Tools > Add-Ins**.
4. Select **Oracle Predictive Analytics** from the Add-Ins dialog box, as shown in the following figure.



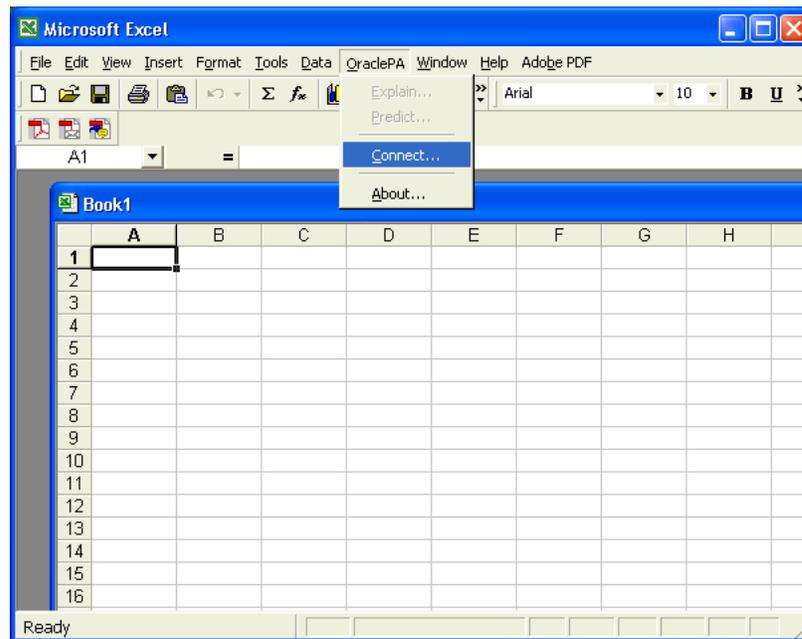
The OraclePA menu is added to the Excel toolbar.

## Connecting Excel to Oracle Database

Before using the add-in, you must have an Oracle Net Service Name for the instance of Oracle Database you plan to use. If Oracle Database is installed on your local PC, then the service name is already defined. You can make a connection to it using the database name. However, if you are connecting to a remote database, then you must define a service name, as described in "[Creating an Oracle Net Service Name](#)" on page 3-3.

To connect to Oracle Database, take these steps:

1. Click **OraclePA > Connect**.

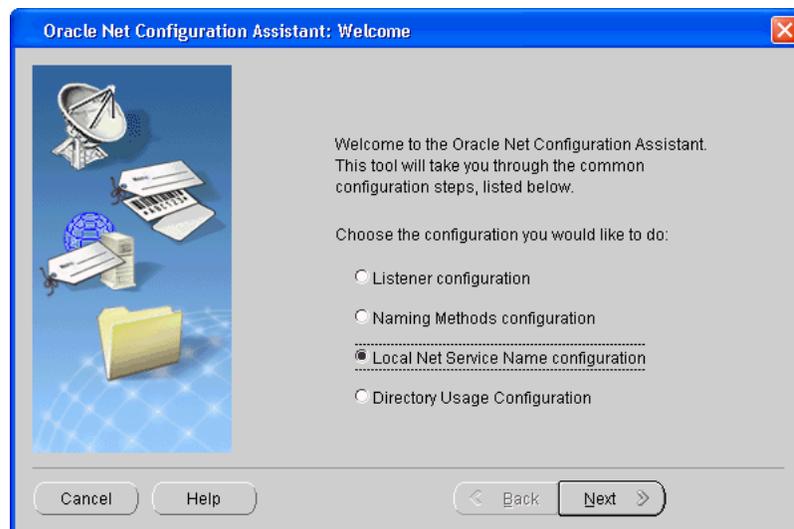


2. On the Connect dialog box, select a service name from the drop-down list, and type in your database user name and password.

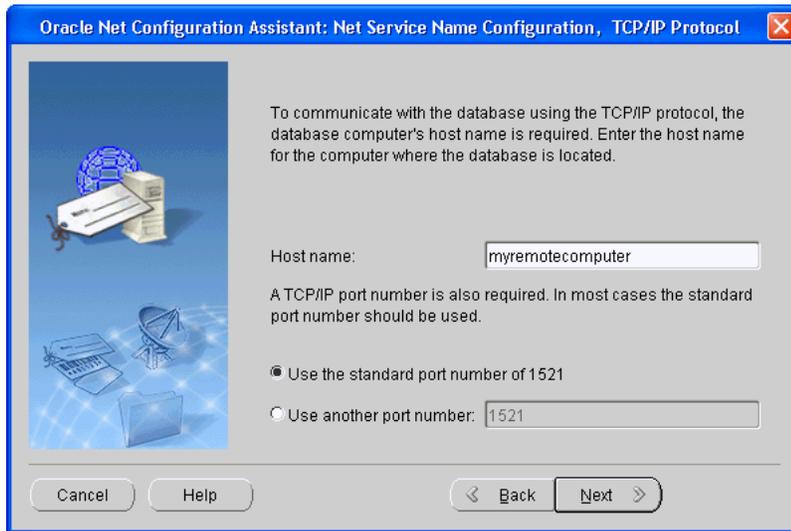
## Creating an Oracle Net Service Name

The following steps explain how to define a simple connection to a remote computer. Depending on your network, you may need to make different choices when running the Net Configuration Assistant wizard. If you installed Oracle Database, then you made many of these choices during installation. Otherwise, ask the DBA for the service name, host computer, and port number.

1. Open Oracle Net Configuration Assistant. Click **Start > All Programs > Oracle - oracle\_home > Configuration and Migration Tools > Net Configuration Assistant**.



2. On the Welcome page, select **Local Net Service Name configuration**.
3. On the Net Service Name Configuration page, select **Add**.
4. On the Service Name page, type the service name for Oracle Database.
5. On the Select Protocols page, select **TCP**.
6. On the TCP/IP Protocol page, provide the name of the host computer where Oracle Database is installed and the port number.



7. On the Test page, select **Yes, perform a test**.
8. On the Connecting page, select **Change Login** and provide your own database user name and password.  
You will not see this page if the default `SYSTEM` user name and password can successfully make a connection.
9. On the Net Service Name page, type a name that uniquely identifies this connection.  
You will choose this name when connecting to Oracle Database from Excel.

## Oracle Data Miner

Oracle Data Miner is a graphical user interface for Oracle Data Mining that helps data analysts find valuable hidden information and patterns in their Oracle data. Oracle Data Miner's easy-to-use wizards guide them through the data preparation, data mining, model evaluation, and model scoring process.

Oracle Data Miner is supported on Windows 2000, Windows XP Professional Edition, and Linux.

## Installing Oracle Data Miner on Windows

To install Oracle Data Miner on a Microsoft Windows platform, take these steps:

1. Download Oracle Data Miner from the Oracle Web site at <http://www.oracle.com/technology/products/bi/odm/index.html>.
2. Open the ZIP file and extract all files to an empty directory (such as C:\ODMINER). Be sure to use folder names so that the files retain their original organization in subfolders.
3. Create a Windows shortcut to BIN\ODMINERW.EXE, and drag the shortcut to your Windows desktop for easy access.

## Installing Oracle Data Miner on Linux

Follow these instructions for installing Oracle Data Miner on Linux.

### Prerequisite

Oracle Data Miner requires Java 1.4.2, which is installed with Oracle Database and Oracle Client. If you are installing Oracle Data Miner on a computer where neither of these products is installed, then check the Java version with this command at the operating system prompt:

```
java -version
```

If Java 1.4.2 is not already installed, you can download it from <http://www.java.com>.

### Installation Steps

To install Oracle Data Miner on a Linux or Unix platform, take these steps:

1. Download Oracle Data Miner from the Oracle Web site at <http://www.oracle.com/technology/products/bi/odm/index.html>.
2. Open the ZIP file and extract all files to an empty directory. The following command creates a directory named `odminer` and inflates the files into it.

```
unzip odminer.zip -d odminer
```

3. Grant execution permission to `bin/odminer` with a command such as this:

```
chmod +x odminer/bin/odminer
```

4. Run the `odminer` executable from the `bin` subdirectory.

```
chmod odminer/bin  
odminer
```

## Connecting Oracle Data Miner to Oracle Database

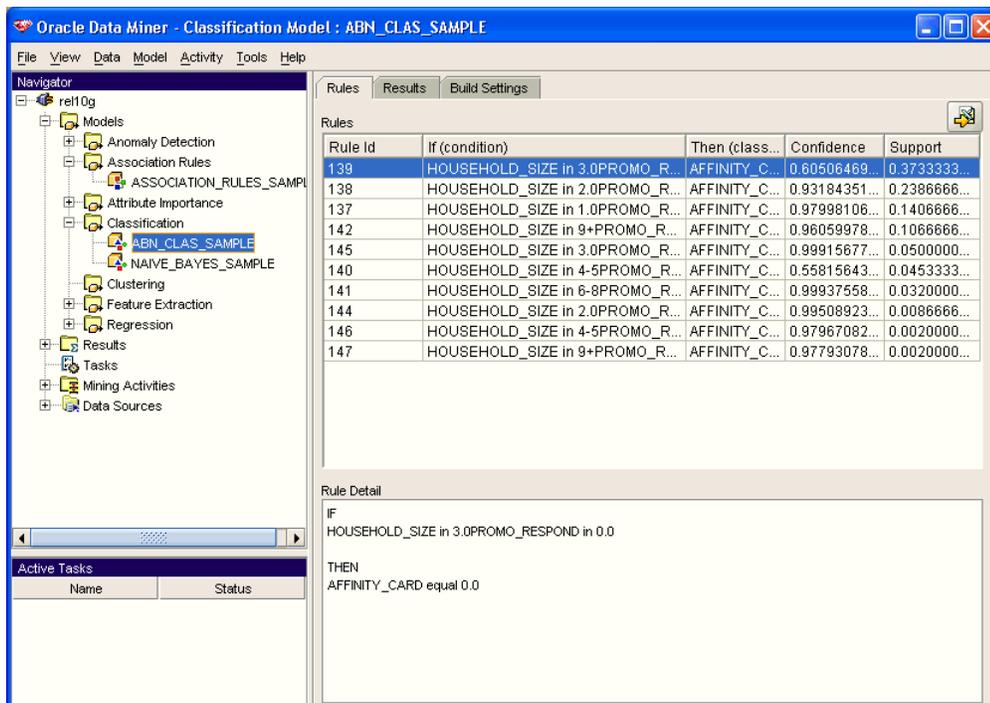
To create a connection from Oracle Data Miner to Oracle Database, take these steps:

1. Open Oracle Data Miner.
2. On the Oracle Data Miner dialog box, click **New**.
3. On the New Connection dialog box, enter the information for connecting to Oracle Database.



The new connection will appear in the Oracle Data Miner dialog box, where you can use it to connect to Oracle Database.

The following figure shows the main page of Oracle Data Miner after opening a new connection.



## Data Mining Application Programming Interfaces

Oracle Database Enterprise Edition provides both a PL/SQL and a Java application programming interface (API) for developing data mining applications. The two APIs provide the facilities to do basic data preparation (such as binning, normalization, winsorizing, clipping, and missing values treatment) and to build, test, and apply models.

The PL/SQL and Java APIs generate models that are interoperable, so that a model can be built with one API and applied using the other API.

**See Also:** *Oracle Data Mining Application Developer's Guide* for more information about the Oracle Data Mining APIs

## PL/SQL Development Tools

The following packages provide data mining functionality to the PL/SQL applications developer:

- DBMS\_DATA\_MINING
- DBMS\_DATA\_MINING\_TRANSFORM
- DBMS\_PREDICTIVE\_ANALYTICS

Several built-in SQL functions support model scoring and deployment.

You can develop applications using either SQL\*Plus or iSQL\*Plus, which can be installed from either the Database or the Client CDs.

### See Also:

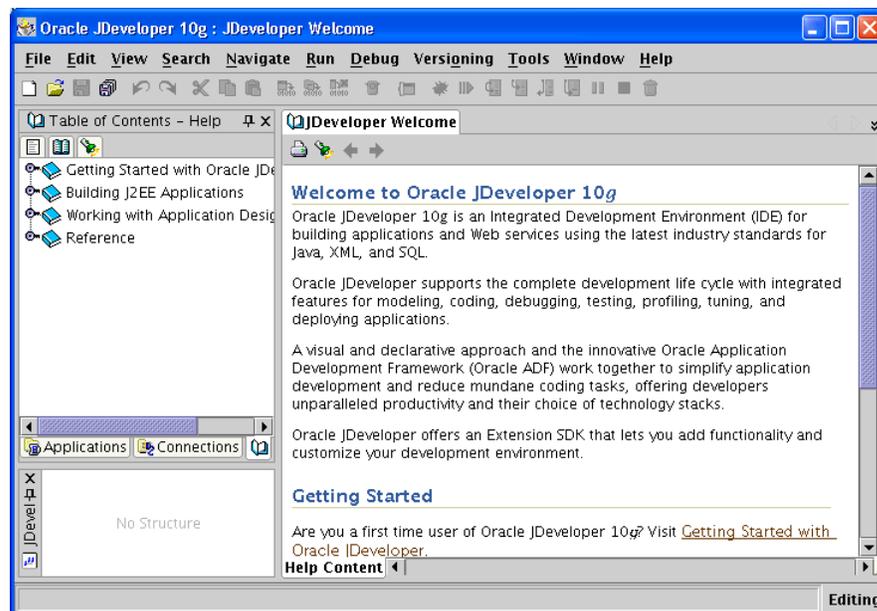
- *Oracle Database PL/SQL Packages and Types Reference* for Data Mining packages
- *Oracle Database SQL Reference* for Data Mining functions

## Java Development Tools

Two Java APIs for Data Mining are provided with Oracle Database:

- JSR-73 Java Data Mining API. Implements the industry standard for data mining.
- Oracle Data Mining Java API. Contains extensions to the JSR-73 standard.

To develop Java applications, you can use a tool such as Oracle JDeveloper, which provides a rich development environment that is fully integrated with Oracle Database. The following screen capture shows the initial Welcome page of JDeveloper.



JDeveloper is an application, and it is not supplied on the Oracle Database CDs. You can download it from the Oracle Technology Network at

<http://www.oracle.com/technology/products/jdev/index.html>.



---

---

## Installing and Using the Sample Programs

A number of sample programs are available with Oracle Data Mining. These programs illustrate the many features of the PL/SQL API, the SQL scoring functions, the Java API, and the BLAST table functions.

The sample programs create a set of models in the database. You can experiment with these models using either the APIs or Oracle Data Miner. You can examine the sample source code, which includes numerous comments, to familiarize yourself with the Oracle Data Mining APIs, and you can create your own models by modifying the samples.

This chapter includes the following sections:

- [Obtaining the Sample Programs](#)
- [Preparing to Run the Sample Programs](#)
- [Examining the Data](#)
- [Using the PL/SQL Samples](#)
- [Using the Java Samples](#)
- [Using the Text Mining Samples](#)
- [Using the BLAST Sample](#)

### Obtaining the Sample Programs

The Oracle Data Mining sample programs are provided with Oracle Database Enterprise Edition. The Companion CD installation process copies the Oracle Data Mining sample programs, along with examples and demos of other database features, to the `\rdbms\demo` subdirectory under `ORACLE_HOME`. For information on installing Oracle Database with the Data Mining option and the sample programs, see [Chapter 1](#).

---

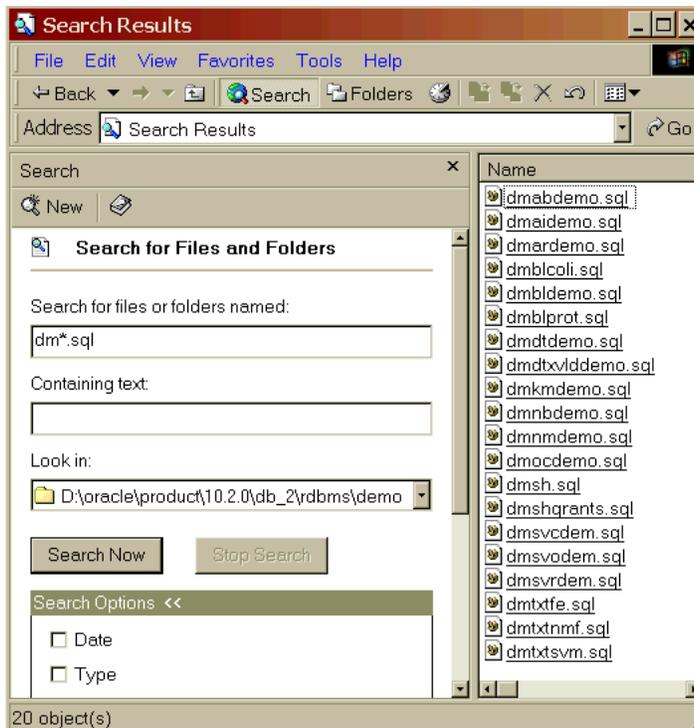
**Note:** The Oracle Data Mining sample programs use data in the SH schema. If the sample schemas were not originally installed in the database, you must install SH.

The sample schema files are copied to `\demo\schema` in the Oracle home directory during installation. To install the sample schemas, run the Database Configuration Assistant. To install SH without the other sample schemas, run the `sh_main` script. The `sh_main.sql` file is located in `\demo\schema\sales_history` in the Oracle home directory. You must run the script as SYS.

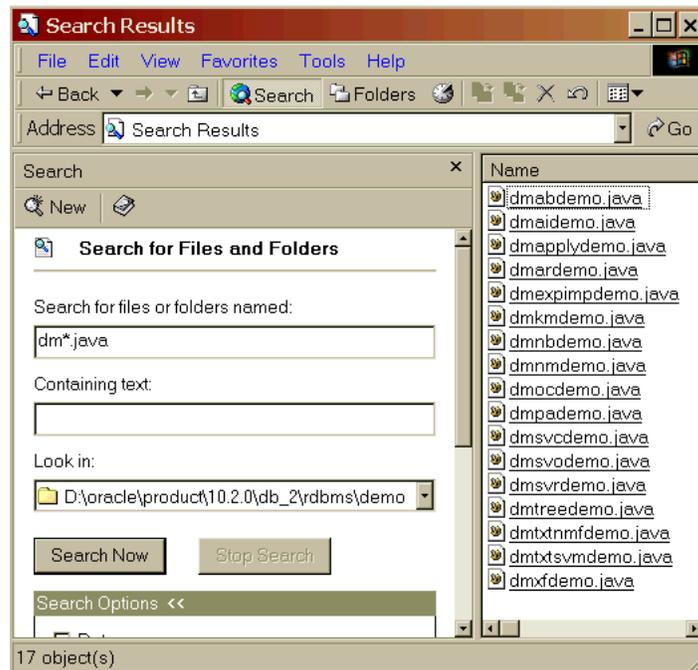
For more details about installing the sample schemas, see *Oracle Database Sample Schemas*.

---

You can list the Oracle Data Mining SQL demo programs in Windows File Manager by searching for `dm*.sql` in `\rdbms\demo` in the Oracle home directory.



You can list the Oracle Data Mining Java demo programs in Windows File Manager by searching for `dm*.java` in `\rdbms\demo` in the Oracle home directory.



## The PL/SQL Samples

The files with a `sql` extension are sample programs that illustrate the PL/SQL API and SQL scoring functions, administrative scripts, and files related to the BLAST algorithm.

Table 4–1 lists the PL/SQL programs that illustrate the mining functions and algorithms supported by Oracle Data Mining.

**Table 4–1 Mining Functions in PL/SQL Samples**

| Mining Function      | Algorithm                                | Sample Program File |
|----------------------|--|---------------------|
| Association Rules    | Apriori                                  | dmardemo.sql        |
| Attribute Importance | Minimum Descriptor Length                | dmaidemo.sql        |
| Classification       | Adaptive Bayes Network                   | dmabdemo.sql        |
| Classification       | Decision Tree                            | dmdtdemo.sql        |
| Classification       | Decision Tree (cross validation)         | dmdtxvlddemo.sql    |
| Classification       | Naive Bayes                              | dmnbdemo.sql        |
| Classification       | Support Vector Machines                  | dmsvcdem.sql        |
| Classification       | Support Vector Machines (One Class)      | dmsvodem.sql        |
| Clustering           | k-Means                                  | dmkmdemo.sql        |
| Clustering           | O-Cluster                                | dmocdemo.sql        |
| Feature Extraction   | Non-Negative Matrix Factorization        | dmnmdemo.sql        |
| Regression           | Support Vector Machines                  | dmsvrdem.sql        |
| Text Mining          | Term extraction using CTX procedures     | dmtxtfe.sql         |
| Text Mining          | Non-Negative Matrix Factorization        | dmtxtnmf.sql        |
| Text Mining          | Support Vector Machines (Classification) | dmtxtsvm.sql        |

Table 4–2 lists the administrative scripts related to the data mining sample programs.

**Table 4–2 Administrative Scripts**

| File Name      | Description  |
|----------------|--|
| dmshgrants.sql | Grants the privileges needed for data mining activities to the data mining user. Also grants SELECT privileges to the data mining user on the tables in the SH schema that contain the data used by the sample programs. |
| dmsh.sql       | In the schema of the data mining user, creates views of the tables in SH that contain the mining data. Also creates tables and indexes that support text mining.   |

Table 4–3 lists the files related to the BLAST algorithm. Instructions for creating sample data sets and running the BLAST sample program are provided in "Using the BLAST Sample" on page 4-16.

**Table 4–3 BLAST Sample Files**

| File Name    | Description   |
|--------------|---|
| dmbldemo.sql | BLAST sample program.   |
| dmblcoli.sql | Script that creates the Ecoli data set.                                   |
| dmblprot.sql | Script that creates the Swiss Protein data set.                           |
| dmblprot.txt | Text for SQL*Loader to load into the Swiss Protein data set.              |
| dmblctl      | Control file for SQL*Loader to use in loading the Swiss Protein data set. |

## The Java Samples

The files with a java extension are sample programs that illustrate the Java API. In addition to the programs that illustrate the basic mining functions, there are programs that illustrate predictive analytics and specific mining tasks, such as data transformations and model import/export.

Table 4–4 lists the Java programs that illustrate the mining functions and algorithms supported by Oracle Data Mining.

**Table 4–4 Mining Functions in Java Samples**

| Mining Function      | Algorithm                           | Sample Program File |
|----------------------|-------------------------------------|---------------------|
| Association Rules    | Apriori                             | dmardemo.java       |
| Attribute Importance | Minimum Descriptor Length           | dmaidemo.java       |
| Classification       | Adaptive Bayes Network              | dmabdemo.java       |
| Classification       | Decision Trees                      | dmtreedemo.java     |
| Classification       | Naive Bayes                         | dmnbdemo.java       |
| Classification       | Support Vector Machines             | dmsvcdemo.java      |
| Classification       | Support Vector Machines (one class) | dmsvodemo.java      |
| Clustering           | K-Means                             | dmkmdemo.java       |
| Clustering           | O-Cluster                           | dmocdemo.java       |
| Feature Extraction   | Non-Negative Matrix Factorization   | dmnmdemo.java       |
| Regression           | Support Vector Machines             | dmsvrdemo.java      |
| Text Mining          | Non-Negative Matrix Factorization   | dmtxtnmfdemo.java   |

**Table 4–4 (Cont.) Mining Functions in Java Samples**

| Mining Function | Algorithm                                | Sample Program File |
|-----------------|--|---------------------|
| Text Mining     | Support Vector Machines (Classification) | dmtxtsvmdemo.java   |

Table 4–5 lists the Java programs that illustrate special mining tasks. These features are all supported in the PL/SQL API as well, since the Java API is layered on the PL/SQL API.

**Table 4–5 Mining Tasks in Java Samples**

| Mining Task                  | Description                          | Sample Program File |
|------------------------------|--------------------------------------|---------------------|
| Data Transformations         | Binning, clipping, and normalization | dmxfdemo.java       |
| Predictive Analytics         | Automated predict and explain        | dmpademo.java       |
| Model Export/Import          | To/from Data Pump dump file          | dmexpimpdemo.java   |
| Classification Model Scoring | Ways of applying an NB model         | dmapplydemo.java    |

## Preparing to Run the Sample Programs

Before attempting to run the sample programs, verify that the following required installation and setup steps have been completed.

1. Oracle Database 10g Enterprise Edition has been installed with the sample schemas according to the instructions in "[Installation of Oracle Database 10g](#)" on page 1-2. If the sample schemas are not installed, follow the instructions in "[Obtaining the Sample Programs](#)" on page 4-1 to install them.
2. The Database Companion CD has been installed according to the instructions in "[Installing the Data Mining Sample Programs](#)" on page 1-7.
3. A data mining user ID has been created according to the instructions in "[Creating a Demo User for Data Mining](#)" on page 2-4. You will use this ID to log in to the database to run the sample programs.
4. The data mining user's schema has been populated with tables and views used by the sample programs according to the instructions in "[Creating a Sample Schema for Data Mining](#)" on page 2-6.
5. You have SQL\*Plus, as described in "[SQL\\*Plus](#)" on page 2-2, for running the PL/SQL programs.
6. You have Java 1.4.2 for running the Java programs. Your CLASSPATH must be set according to the instructions in "[Using the Java Samples](#)" on page 4-12.
7. You can connect to the database as the data mining user.

### A Sample User Scenario

Let's assume that you have access to Oracle Database 10g, which has been installed on a Windows host with the Data Mining option, the sample schemas, and the sample programs. You can log in to this database as SYS using SQL\*Plus.

Several Data Mining users (dmuser1 and dmuser2) have already been created in this database. You want to create your own user ID (dmuser3) and run the sample programs to create the sample models in your schema. To accomplish this, you would perform the following steps.

1. Log in to the database as *SYS* and create the *dmuser3* user.

```
> sqlplus
Enter user-name: sys / as sysdba
Enter password: sys_password
SQL> create user dmuser3 identified by dmuser3_password
      default tablespace users temporary tablespace temp
      quota unlimited on users;
```

2. Run *dmshgrants.sql* to grant privileges to *dmuser3*. These privileges are required for performing data mining tasks and for accessing the mining data in *SH* used by the sample programs.

```
SQL> @ %ORACLE_HOME%\rdbms\demo\dmshgrants SH_password dmuser3
```

3. Connect as *dmuser3* and run *dmsh.sql* to populate the *dmuser3* schema with views and tables needed by the sample programs. Save your changes in the database.

```
SQL> connect dmuser3/dmuser3_password
SQL> @ %ORACLE_HOME%\rdbms\demo\dmsh
SQL> commit;
```

You can now run any of the PL/SQL data mining samples as described in ["Using the PL/SQL Samples"](#) on page 4-10. For example, while logged in to SQL\*Plus as *dmuser3*, you could run the Naive Bayes PL/SQL sample with the following statement.

```
SQL>@ %ORACLE_HOME%\rdbms\demo\dmnbdemo
```

You can also run any of the Java data mining samples, if you have Java 1.4.2 or higher and your *CLASSPATH* is set as described in ["Using the Java Samples"](#) on page 4-12. For example, you could run the Naive Bayes Java sample with a command like the following at the operating system prompt.

```
>java dmnbdemo myserver:1521:orcl dmuser3 dmuser3_password
```

Note that the BLAST demo uses different data sets and requires a separate setup procedure. See ["Using the BLAST Sample"](#) on page 4-16.

---



---

**Note:** All the demo programs are re-executable. They start by deleting the results of the previous run before executing the current run.

---



---

## Examining the Data

The *dmsh.sql* script creates views, tables, and indexes in the user's schema. The views define columns of customer data from tables in the *SH* schema. This data is used by the sample data mining programs. The tables reference the same columns in *SH*, but they include an extra *COMMENTS* column for text mining. The indexes are used to extract terms from the text in the *COMMENTS* column and build a nested table column.

### Customer Data for Data Mining

Views in the data mining user's schema define columns of data from the *CUSTOMERS*, *SALES*, *PRODUCTS*, *COUNTRIES*, and *SUPPLEMENTARY\_DEMOGRAPHICS* tables in the *SH* schema. You can list these views with the following SQL statements.

```
SQL>connect dmuser3/dmuser3_password
```

```
SQL>select view_name from user_views;
```

The views are listed in [Table 4–6](#).

**Table 4–6 Views Used by the Data Mining Sample Programs**

| View Name               | Description                                    |
|-------------------------|--|
| MINING_DATA_APPLY_STR_V | Scoring data for o-cluster                     |
| MINING_DATA_BUILD_STR_V | Build data for o-cluster                       |
| MINING_DATA_APPLY_V     | Scoring data for data mining (not text mining) |
| MINING_DATA_BUILD_V     | Build data for data mining (not text mining)   |
| MINING_DATA_TEST_V      | Test data for data mining (not text mining)    |
| MARKET_BASKET_V         | Data for association rules                     |
| MINING_DATA_ONE_CLASS_V | Data for one-class SVM                         |

You can see the references to tables in SH by listing the view definitions. The definition of the view MINING\_DATA\_BUILD\_V is shown as follows.

```
SQL>set long 1000000
SQL>set longc 100000
SQL>select text from all_views where
       owner='DMUSER3'and view_name='MINING_DATA_BUILD_V';

      SELECT a.CUST_ID, a.CUST_GENDER, 2003-a.CUST_YEAR_OF_BIRTH AGE,
             a.CUST_MARITAL_STATUS, c.COUNTRY_NAME, a.CUST_INCOME_LEVEL,
             b.EDUCATION, b.OCC UPATION, b.HOUSEHOLD_SIZE, b.YRS_RESIDENCE,
             b.AFFINITY_CARD, b.BULK_PACK_DISKETTES, b.FLAT_PANEL_MONITOR,
             b.HOME_THEATER_PACKAGE, b.BOOKKEEPING_APPLICATION,
             b.PRINTER_SUPPLIES, b.Y_BOX_GAMES, b.OS_DOC_SET_KANJI
      FROM   sh.customers a,
             sh.supplementary_demographics b,
             sh.countries c
      WHERE  a.CUST_ID = b.CUST_ID AND a.country_id = c.country_id
             AND a.cust_id between 101501 and 103000
```

The views are used to build, test, and score the sample models. Each view has a CUSTOMER\_ID column, which is the case ID, and an AFFINITY\_CARD column, which is the target used by the predictive models. Most of the views provide data for 1500 customers (1500 rows). The view used by the one-class SVM model has data for 940 customers.

The columns of build data in the MINING\_DATA\_BUILD\_V view are listed in the following example.

```
SQL> describe MINING_DATA_BUILD_V

CUST_ID                NOT NULL                NUMBER
CUST_GENDER            NOT NULL                CHAR(1)
AGE                    NOT NULL                NUMBER
CUST_MARITAL_STATUS   NOT NULL                VARCHAR2(20)
COUNTRY_NAME           NOT NULL                VARCHAR2(40)
CUST_INCOME_LEVEL     NOT NULL                VARCHAR2(30)
EDUCATION              NOT NULL                VARCHAR2(21)
OCCUPATION             NOT NULL                VARCHAR2(21)
HOUSEHOLD_SIZE        NOT NULL                VARCHAR2(21)
YRS_RESIDENCE         NOT NULL                NUMBER
AFFINITY_CARD         NOT NULL                NUMBER(10)
```

|                         |            |
|-------------------------|------------|
| BULK_PACK_DISKETTES     | NUMBER(10) |
| FLAT_PANEL_MONITOR      | NUMBER(10) |
| HOME_THEATER_PACKAGE    | NUMBER(10) |
| BOOKKEEPING_APPLICATION | NUMBER(10) |
| PRINTER_SUPPLIES        | NUMBER(10) |
| Y_BOX_GAMES             | NUMBER(10) |
| OS_DOC_SET_KANJI        | NUMBER(10) |

## Market Basket Data for Association Rules

The association rules samples use the MARKET\_BASKET\_V data set, which includes columns of products from the PRODUCTS table and the CUSTOMER\_ID column from the CUSTOMERS table in SH. The columns of the MARKET\_BASKET\_V view are listed in the following example.

```
SQL> describe MARKET_BASKET_V
```

|                             |          |        |
|-----------------------------|----------|--------|
| CUST_ID                     | NOT NULL | NUMBER |
| EXTENSION_CABLE             |          | NUMBER |
| FLAT_PANEL_MONITOR          |          | NUMBER |
| CD_RW_HIGH_SPEED_5_PACK     |          | NUMBER |
| ENVOY_256MB_40GB            |          | NUMBER |
| ENVOY_AMBASSADOR            |          | NUMBER |
| EXTERNAL_8X_CD_ROM          |          | NUMBER |
| KEYBOARD_WRIST_REST         |          | NUMBER |
| SM26273_BLACK_INK_CARTRIDGE |          | NUMBER |
| MOUSE_PAD                   |          | NUMBER |
| MULTIMEDIA_SPEAKERS_3INCH   |          | NUMBER |
| OS_DOC_SET_ENGLISH          |          | NUMBER |
| SIMM_16MB_PCMCIAII_CARD     |          | NUMBER |
| STANDARD_MOUSE              |          | NUMBER |

## Customer Data for Text Mining

The text mining samples use the same customer data from tables in SH, but they include either an extra text column or a collection type column. The collection type is a nested table of type DM\_NESTED\_NUMERICALS.

You can list these tables with the following SQL statements.

```
SQL>connect dmuser3/dmuser3_password
SQL>select table_name from user_tables where table_name like '%MINING%';
```

The tables are listed in [Table 4-7](#).

**Table 4-7 Tables Used by the Sample Text Mining Programs**

| Table Name               | Description  |
|--------------------------|--|
| MINING_APPLY_NESTED_TEXT | Apply table with COMMENTS column as DM_NESTED_NUMERICALS |
| MINING_BUILD_NESTED_TEXT | Build table with COMMENTS column as DM_NESTED_NUMERICALS |
| MINING_TEST_NESTED_TEXT  | Test table with COMMENTS column as DM_NESTED_NUMERICALS  |
| MINING_APPLY_TEXT        | Apply table with COMMENTS column as VARCHAR2 (4000)      |
| MINING_BUILD_TEXT        | Build table with COMMENTS column as VARCHAR2 (4000)      |
| MINING_TEST_TEXT         | Test table with COMMENTS column as VARCHAR2 (4000)       |

In the MINING\_BUILD\_TEXT, MINING\_TEST\_TEXT, and MINING\_APPLY\_TEXT tables, the COMMENTS column is of type VARCHAR2 (4000).

```
SQL> describe MINING_BUILD_TEXT
```

| Name                    | Null?    | Type           |
|-------------------------|----------|----------------|
| CUST_ID                 | NOT NULL | NUMBER         |
| CUST_GENDER             | NOT NULL | CHAR(1)        |
| AGE                     |          | NUMBER         |
| CUST_MARITAL_STATUS     |          | VARCHAR2(20)   |
| COUNTRY_NAME            | NOT NULL | VARCHAR2(40)   |
| CUST_INCOME_LEVEL       |          | VARCHAR2(30)   |
| EDUCATION               |          | VARCHAR2(21)   |
| OCCUPATION              |          | VARCHAR2(21)   |
| HOUSEHOLD_SIZE          |          | VARCHAR2(21)   |
| YRS_RESIDENCE           |          | NUMBER         |
| AFFINITY_CARD           |          | NUMBER(10)     |
| BULK_PACK_DISKETTES     |          | NUMBER(10)     |
| FLAT_PANEL_MONITOR      |          | NUMBER(10)     |
| HOME_THEATER_PACKAGE    |          | NUMBER(10)     |
| BOOKKEEPING_APPLICATION |          | NUMBER(10)     |
| PRINTER_SUPPLIES        |          | NUMBER(10)     |
| Y_BOX_GAMES             |          | NUMBER(10)     |
| OS_DOC_SET_KANJI        |          | NUMBER(10)     |
| COMMENTS                |          | VARCHAR2(4000) |

In the MINING\_\*\_NESTED\_TEXT tables, the COMMENTS column is of type DM\_NESTED\_NUMERICALS.

```
SQL> describe MINING_BUILD_NESTED_TEXT
```

| Name                    | Null?    | Type                       |
|-------------------------|----------|----------------------------|
| CUST_ID                 | NOT NULL | NUMBER                     |
| CUST_GENDER             | NOT NULL | CHAR(1)                    |
| AGE                     |          | NUMBER                     |
| CUST_MARITAL_STATUS     |          | VARCHAR2(20)               |
| COUNTRY_NAME            | NOT NULL | VARCHAR2(40)               |
| CUST_INCOME_LEVEL       |          | VARCHAR2(30)               |
| EDUCATION               |          | VARCHAR2(21)               |
| OCCUPATION              |          | VARCHAR2(21)               |
| HOUSEHOLD_SIZE          |          | VARCHAR2(21)               |
| YRS_RESIDENCE           |          | NUMBER                     |
| AFFINITY_CARD           |          | NUMBER(10)                 |
| BULK_PACK_DISKETTES     |          | NUMBER(10)                 |
| FLAT_PANEL_MONITOR      |          | NUMBER(10)                 |
| HOME_THEATER_PACKAGE    |          | NUMBER(10)                 |
| BOOKKEEPING_APPLICATION |          | NUMBER(10)                 |
| PRINTER_SUPPLIES        |          | NUMBER(10)                 |
| Y_BOX_GAMES             |          | NUMBER(10)                 |
| OS_DOC_SET_KANJI        |          | NUMBER(10)                 |
| COMMENTS                |          | DMSYS.DM_NESTED_NUMERICALS |

The process of extracting terms from a text column into a nested table column is described in ["Using the Text Mining Samples"](#) on page 4-14 and in the *Oracle Data Mining Application Developer's Guide*.

## Using the PL/SQL Samples

The PL/SQL samples illustrate the use of the `DBMS_DATA_MINING` package for creating models and the `DBMS_DATA_MINING_TRANSFORM` package for performing transformations on the mining data. Several of the models include examples of the SQL scoring functions for applying models within a SQL statement. The PL/SQL sample programs are listed in [Table 4-1](#).

**See Also:** *Oracle Database PL/SQL Packages and Types Reference* and *Oracle Data Mining Application Developer's Guide* for information on the Oracle Data Mining PL/SQL API. See *Oracle Database SQL Reference* for information on the SQL functions for data mining.

## Running the PL/SQL Programs

In SQL\*Plus, use commands like the following to execute the sample programs and list the models created by them.

```
>sqlplus dmuser3/dmuser3_password
SQL> set serveroutput on
SQL> set echo on
SQL> @ %ORACLE_HOME%\rdbms\demo\program_name
.
.
.
.
SQL> set linesize 200
SQL> set pagesize 100
SQL> select NAME, FUNCTION_NAME, ALGORITHM_NAME, TARGET_ATTRIBUTE from DM_USER_MODELS;
```

| NAME                | FUNCTION_NAME        | ALGORITHM_NAME             | TARGET_ATTRIBUTE |
|---------------------|----------------------|----------------------------|------------------|
| T_NMF_SAMPLE        | FEATURE_EXTRACTION   | NONNEGATIVE_MATRIX_FACTOR  |                  |
| T_SVM_CLAS_SAMPLE   | CLASSIFICATION       | SUPPORT_VECTOR_MACHINES    | AFFINITY_CARD    |
| AR_SH_SAMPLE        | ASSOCIATION_RULES    | APRIORI_ASSOCIATION_RULES  |                  |
| AI_SH_SAMPLE        | ATTRIBUTE_IMPORTANCE | MINIMUM_DESCRIPTION_LENGTH | AFFINITY_CARD    |
| ABN_SH_CLAS_SAMPLE  | CLASSIFICATION       | ADAPTIVE_BAYES_NETWORK     | AFFINITY_CARD    |
| DT_SH_CLAS_SAMPLE   | CLASSIFICATION       | DECISION_TREE              | AFFINITY_CARD    |
| NB_SH_CLAS_SAMPLE   | CLASSIFICATION       | NAIVE_BAYES                | AFFINITY_CARD    |
| SVMC_SH_CLAS_SAMPLE | CLASSIFICATION       | SUPPORT_VECTOR_MACHINES    | AFFINITY_CARD    |
| OC_SH_CLUS_SAMPLE   | CLUSTERING           | O_CLUSTER                  |                  |
| KM_SH_CLUS_SAMPLE   | CLUSTERING           | KMEANS                     |                  |
| NMF_SH_SAMPLE       | FEATURE_EXTRACTION   | NONNEGATIVE_MATRIX_FACTOR  |                  |
| SVMR_SH_REGR_SAMPLE | REGRESSION           | SUPPORT_VECTOR_MACHINES    | AGE              |

## PL/SQL Program Summaries

Summary descriptions of the PL/SQL sample programs are provided in [Table 4-8](#). The programs are listed individual by name and function in [Table 4-1](#).

For detailed descriptions of the sample programs, see the comments in the source code.

**Table 4–8 Overview of the PL/SQL Sample Programs**

| Mining Function      | Description   |
|----------------------|---|
| Classification       | <p>The classification programs demonstrate various preprocessing techniques and perform the following steps:</p> <ul style="list-style-type: none"> <li>■ Build a classification model using training data</li> <li>■ Display model details and settings</li> <li>■ Test the model by applying the model on the test data</li> <li>■ Present test metrics, such as confusion matrix, lift, and ROC</li> <li>■ Apply the model on the scoring data</li> <li>■ Present apply results</li> <li>■ Present ranked apply results, influenced by a cost matrix</li> </ul> <p>The <code>dmtdtxv1ddemo.sql</code> program demonstrates cross-validation techniques for decision tree based-classification. With minor modifications, this program can be used to perform cross validation using other models/algorithms.</p> |
| Regression           | <code>dmsvrddem.sql</code> uses different test metrics, but otherwise performs most of the same steps used in the classification programs. Selected attributes of the input data are preprocessed (normalized).   |
| Association          | <code>dmardemo.sql</code> builds an association model and presents frequent itemsets and association rules as output. Selected attributes of the input data are preprocessed (binned).  |
| Clustering           | <code>dmkmdemo.sql</code> ( <i>k</i> -Means) and <code>dmocdemo.sql</code> (0-cluster) build clustering models and present cluster details, such as rules, centroid, and histogram for each cluster as output. The models are scored, and the probabilities associated with each cluster are returned as output. Selected attributes of the input data are preprocessed (normalized).   |
| Feature Extraction   | <code>dmnmddemo.sql</code> builds a feature extraction model and presents model details as the output. The model is scored, and each feature ID is associated with a probability. Selected attributes of the input data are preprocessed (normalized).  |
| Attribute Importance | <code>dmaiddemo.sql</code> builds an attribute importance model and presents a list of important attributes as the output of model details. Selected attributes of the input data are preprocessed (binned).  |

## SQL Scoring Functions

Some of the PL/SQL sample programs use SQL scoring functions to apply models created with the `DBMS_DATA_MINING` package. The SQL scoring functions can also be used to apply models created with the Java API.

The programs that demonstrate the SQL scoring functions are listed in [Table 4–9](#).

**Table 4–9 SQL Scoring Functions in the Sample Programs**

| Program Name               | Algorithm       | SQL Functions Used                               |
|----------------------------|-----------------|--|
| <code>dmkmdemo.sql</code>  | <i>k</i> -Means | CLUSTER_ID<br>CLUSTER_PROBABILITY<br>CLUSTER_SET |
| <code>dmocdemo.sql</code>  | O-Cluster       | CLUSTER_ID                                       |
| <code>dmnmddemo.sql</code> | NMF             | FEATURE_ID<br>FEATURE_SET<br>FEATURE_VALUE       |

**Table 4–9 (Cont.) SQL Scoring Functions in the Sample Programs**

| Program Name | Algorithm          | SQL Functions Used  |
|--------------|--------------------|---|
| dmdtdemo.sql | Decision Tree      | PREDICTION<br>PREDICTION_COST<br>PREDICTION_DETAILS<br>PREDICTION_SET |
| dmsvcdem.sql | SVM Classification | PREDICTION<br>PREDICTION_PROBABILITY<br>PREDICTION_SET                |
| dmsvodem.sql | SVM One Class      | PREDICTION<br>PREDICTION_PROBABILITY                                  |
| dmsvrдем.sql | SVM Regression     | PREDICTION  |
| dmtxtsvm.sql | Text Mining        | PREDICTION<br>PREDICTION_PROBABILITY                                  |

## Using the Java Samples

The Java samples illustrate the features of the Oracle Data Mining Java API, which implements Oracle-specific extensions to the Java Data Mining (JDM) 1.0 standard. The Java sample programs are listed in [Table 4–4](#).

The Java samples demonstrate data preprocessing and the basic mining functions. Additional Java samples demonstrate predictive analytics, import/export, and text mining. Predictive analytics are implemented in PL/SQL as the DBMS\_PREDICTIVE\_ANALYTICS package; predictive analytics can be accessed in a spreadsheet environment with the Oracle Spreadsheet Add-In for Predictive Analytics. Model import/export is implemented in PL/SQL as the IMPORT\_MODEL and EXPORT\_MODEL procedures in the DBMS\_DATA\_MINING package.

Text mining in PL/SQL uses Oracle Text routines; the process is summarized in "[Text Mining in PL/SQL](#)" on page 4-15. Text mining in Java is described in "[Text Mining in Java](#)" on page 4-16. The Java text mining samples use the data that has not undergone preprocessing and does not contain nested tables. The data used by the text mining samples is described in "[Customer Data for Text Mining](#)" on page 4-8.

**See Also:** *Oracle Data Mining Java API Reference* (javadoc) and the *Oracle Data Mining Application Developer's Guide* for information on the Java API.

## Preparing to Run the Java Programs

Before running the Java programs, do the following:

1. Check that the version of Java you are using is 1.4.2. You can execute the following in a command window to check the version of Java:

```
>java -version
```

2. Ensure that your CLASSPATH includes the following JAR files:

```
%ORACLE_HOME%\rdbms\jlib\jdm.jar
%ORACLE_HOME%\rdbms\jlib\ojdm_api.jar
%ORACLE_HOME%\rdbms\jlib\xdb.jar
%ORACLE_HOME%\jdbc\lib\ojdbc14.jar
%ORACLE_HOME%\oc4j\j2ee\home\lib\connector.jar
%ORACLE_HOME%\jlib\orai18n.jar
```

```
%ORACLE_HOME%\jlib\orai18n-mapping.jar
%ORACLE_HOME%\lib\xmlparserv2.jar
```

3. Compile the programs. Use the `javac` executable in the `\jdk\bin\` directory within Oracle home.

```
>javac program_name
```

## Running the Java Programs

Use the following syntax to execute the sample Java programs. Use the `java` executable in the `\jdk\bin\` directory within Oracle home.

```
>java program_name connect_string user_name user_password
```

The connection string specifies your Oracle database connection. It identifies the machine hosting the database, the port through which the connection is made, and the name of the database instance (the Oracle system identifier).

```
host_name:port:SID
```

For example, the following command executes the Naive Bayes sample `dmnbdemo.java` as `dmuser3` in the database instance `orcl` on host `mach05` at port `1521`.

```
>java dmnbdemo mach05:1521:orcl dmuser3 dmuser3_password
```

You can list the models created by the Java programs with commands like the following in SQL\*Plus.

```
>sqlplus dmuser3/dmuser3_password
SQL> set linesize 200
SQL> set pagesize 100
SQL> select NAME, FUNCTION_NAME, ALGORITHM_NAME, TARGET_ATTRIBUTE
       from dm_user_models where NAME like '%JDM';
```

| NAME            | FUNCTION_NAME        | ALGORITHM_NAME             | TARGET_ATTRIBUTE |
|-----------------|----------------------|----------------------------|------------------|
| TXTNMFMODEL_JDM | FEATURE_EXTRACTION   | NONNEGATIVE_MATRIX_FACTOR  |                  |
| ARMODEL_JDM     | ASSOCIATION_RULES    | APRIORI_ASSOCIATION_RULES  |                  |
| TREEMODEL_JDM   | CLASSIFICATION       | DECISION_TREE              | AFFINITY_CARD    |
| AIMODEL_JDM     | ATTRIBUTE_IMPORTANCE | MINIMUM_DESCRIPTION_LENGTH | AFFINITY_CARD    |
| ABNMODEL_JDM    | CLASSIFICATION       | ADAPTIVE_BAYES_NETWORK     | AFFINITY_CARD    |
| NBMODEL_JDM     | CLASSIFICATION       | NAIVE_BAYES                | AFFINITY_CARD    |
| SVMCMODEL_JDM   | CLASSIFICATION       | SUPPORT_VECTOR_MACHINES    | AFFINITY_CARD    |
| SVMOMODEL_JDM   | CLASSIFICATION       | SUPPORT_VECTOR_MACHINES    |                  |
| KMMODEL_JDM     | CLUSTERING           | KMEANS                     |                  |
| OCMODEL_JDM     | CLUSTERING           | O_CLUSTER                  |                  |
| NMFMODEL_JDM    | FEATURE_EXTRACTION   | NONNEGATIVE_MATRIX_FACTOR  |                  |
| SVMRMODEL_JDM   | REGRESSION           | SUPPORT_VECTOR_MACHINES    | AGE              |
| TXTSVMODEL_JDM  | CLASSIFICATION       | SUPPORT_VECTOR_MACHINES    | AFFINITY_CARD    |

## Java Program Summaries

Summary descriptions of the Java sample programs are provided in [Table 4-10](#). The programs are listed individually by name in [Table 4-4](#) and [Table 4-5](#).

For detailed descriptions of the sample programs, see the comments in the source code.

**Table 4–10 Overview of the Java Sample Programs**

| Mining Function or Task | Description  |
|-------------------------|--|
| Classification          | <p>The classification programs demonstrate various preprocessing techniques and perform the following steps:</p> <ul style="list-style-type: none"> <li>■ Build a classification model using training data</li> <li>■ Display model details and settings</li> <li>■ Test the model by applying the model on the test data</li> <li>■ Present test metrics, such as confusion matrix, lift, and ROC</li> <li>■ Apply the model on the scoring data</li> <li>■ Present apply results</li> <li>■ Present ranked apply results, influenced by a cost matrix</li> </ul> <p>The <code>dmapplydemo.java</code> program demonstrates several ways of applying a Naive Bayes model.</p> |
| Regression              | <p><code>dmsvrdemo.java</code> uses different test metrics, but otherwise performs most of the same steps used in the classification programs. Selected attributes of the input data are preprocessed (normalized).</p>  |
| Association             | <p><code>dmardemo.java</code> builds an association model and presents frequent itemsets and association rules as output. Selected attributes of the input data are preprocessed (binned).</p>   |
| Clustering              | <p><code>dmkmdemo.java</code> (<i>k</i>-Means) and <code>dmocdemo.java</code> (0-cluster) build clustering models and present cluster details, such as rules, centroid, and histogram for each cluster as output. The models are scored, and the probabilities associated with each cluster are returned as output. Selected attributes of the input data are preprocessed (normalized).</p>   |
| Feature Extraction      | <p><code>dmnmdemo.java</code> builds a feature extraction model and presents model details as the output. The model is scored, and each feature ID is associated with a probability. Selected attributes of the input data are preprocessed (normalized).</p>  |
| Attribute Importance    | <p><code>dmaidemo.java</code> builds an attribute importance model and presents a list of important attributes as the output of model details. Selected attributes of the input data are preprocessed (binned).</p>  |
| Data Transformations    | <p><code>dmxfdemo.java</code> demonstrates binning, clipping, and normalization transformations.</p>   |
| Predictive Analytics    | <p><code>dmpademo.java</code> demonstrates PREDICT and EXPLAIN functions.</p>  |
| Model Import/Export     | <p><code>dmexpimpdemo.java</code> builds a Naive Bayes model, exports it to a dump file, then imports it from the dump file.</p>   |

## Using the Text Mining Samples

Oracle Data Mining can mine text columns that have undergone pre-processing by Oracle Text routines.

Oracle Text is a technology for building text query and document classification applications. It provides indexing, word and theme searching, and viewing capabilities for text. Oracle Text is included in a general installation of Oracle Database Enterprise Edition, and therefore is already present in a database installed according to the instructions in [Chapter 1](#).

The pre-processing steps for text mining create nested table columns of type `DM_NESTED_NUMERICALS` from columns of type `VARCHAR2` or `CLOB`. Each row of the nested table specifies an attribute name and a value. The type definition is as follows.

```
CREATE OR REPLACE TYPE DM_NESTED_NUMERICAL AS OBJECT
  (attribute_name VARCHAR2(30),
   value          NUMBER)
/
CREATE OR REPLACE TYPE DM_NESTED_NUMERICALS AS TABLE OF DM_NESTED_NUMERICAL
```

Terms extracted from text documents into nested tables can become generic attributes in training or scoring data. Classification, clustering, and feature-extraction models can be built using these attributes.

Sample text mining programs in both PL/SQL and Java illustrate classification and feature extraction of a pre-processed text column.

## Text Mining in PL/SQL

Three PL/SQL sample programs illustrate the process of text mining. One program illustrates the pre-processing that is required to prepare the data for mining. The other two programs build models that use the transformed text.

### Text Transformation Sample

To prepare a column for text mining using the PL/SQL API, you must use Oracle Text routines to perform the following general steps:

1. Create a domain index on the column.
2. Use the index to extract terms from the column to a temporary table.
3. Populate a column of type `DM_NESTED_NUMERICALS` with the terms in the temporary table.

The process of term extraction using Oracle Text is illustrated in the sample program `dmtxtfe.sql`. The source code contains extensive comments that explain the steps involved in transforming text into a set of features that can be mined using Oracle Data Mining.

More details about text transformation are provided in the *Oracle Data Mining Application Developer's Guide*.

### Text Transformation for the PL/SQL Sample Text Mining Programs

The `dmsb.sql` script performs the text transformation required by the PL/SQL text mining samples. There are two such samples: `dmtxtnmf.sql`, which builds a feature extraction model using Non-Negative Matrix Factorization, and `dmtxtsvm.sql`, which builds a classification model using Support Vector Machines. Both of these programs use the tables that have a nested table column of comment data.

```
MINING_BUILD_NESTED_TEXT
MINING_TEST_NESTED_TEXT
MINING_APPLY_NESTED_TEXT
```

### The Sample Text Mining Models (PL/SQL)

You can run the PL/SQL text mining sample programs, `dmtxtnmf.sql` and `dmtxtsvm.sql`, after simply following the instructions in "[Preparing to Run the Sample Programs](#)" on page 4-5. The models created by these programs are listed in the following example.

```
SQL> @ %ORACLE_HOME%\rdbms\demo\dmtxtnmf.sql
SQL> @ %ORACLE_HOME%\rdbms\demo\dmtxsvm.sql
SQL> select NAME, FUNCTION_NAME, ALGORITHM_NAME, TARGET_ATTRIBUTE
        from dm_user_models;
```

| NAME              | FUNCTION_NAME      | ALGORITHM_NAME            | TARGET_ATTRIBUTE |
|-------------------|--------------------|---------------------------|------------------|
| T_NMF_SAMPLE      | FEATURE_EXTRACTION | NONNEGATIVE_MATRIX_FACTOR |                  |
| T_SVM_CLAS_SAMPLE | CLASSIFICATION     | SUPPORT_VECTOR_MACHINES   | AFFINITY_CARD    |

## Text Mining in Java

Two Java sample programs illustrate the process of text mining. One builds a feature extraction model, the other builds a classification model.

### Text Transformation for the Java Sample Text Mining Programs

The Oracle Data Mining Java API provides an interface that handles the term extraction process. If you are developing data mining applications in Java, you do not need to use Oracle Text directly. However, you must ensure that Oracle Text is present in the database.

The `OraTextTransform` interface is used to perform text transformation within the Java text mining samples. There are two such samples: `dmtxtnmfdemo.java`, which builds a feature extraction model using Non-Negative Matrix Factorization, and `dmtxsvmdemo.java`, which builds a classification model using Support Vector Machines. Both of these programs create build, test, and apply data sets from the tables that have a text column of comment data.

```
MINING_BUILD_TEXT
MINING_TEST_TEXT
MINING_APPLY_TEXT
```

### The Sample Text Mining Models (Java)

You can run the Java text mining sample programs, `dmtxtnmfdemo.java` and `dmtxsvmdemo.java`, after following the instructions in "[Preparing to Run the Sample Programs](#)" on page 4-5. The models created by these programs are shown in the following example.

```
> java dmtxtnmfdemo host:port:SID dmuser3 dmuser3_password
> java dmtxsvmdemo host:port:SID dmuser3 dmuser3_password
>sqlplus dmuser3/dmuser3_password
SQL> select NAME, FUNCTION_NAME, ALGORITHM_NAME, TARGET_ATTRIBUTE
        from dm_user_models;
```

| NAME            | FUNCTION_NAME      | ALGORITHM_NAME            | TARGET_ATTRIBUTE |
|-----------------|--------------------|---------------------------|------------------|
| txtnmfModel_jdm | FEATURE_EXTRACTION | NONNEGATIVE_MATRIX_FACTOR |                  |
| txtsvmModel_jdm | CLASSIFICATION     | SUPPORT_VECTOR_MACHINES   | AFFINITY_CARD    |

## Using the BLAST Sample

The Oracle implementation of the Basic Local Alignment Search Tool (BLAST) is demonstrated in the sample program, `dmbldemo.sql`. This program provides examples of sequence matching queries using the BLAST table functions.

The BLAST sample program and configuration scripts are listed in [Table 4-3, "BLAST Sample Files"](#).

**See Also:** *Oracle Data Mining Application Developer's Guide* and *Oracle Data Mining Concepts* for information on BLAST.

## Preparing to Run the BLAST Demo

The sample BLAST table functions in `dmbldemo.sql` use two data sets: `SwissProt` and `ecoli10`. To prepare these data sets, log in to SQL\*Plus as the data mining user and run the `dmb1prot` and `dmb1coli` scripts as shown in the following example.

```
SQL>connect dmuser3/dmuser3_password
SQL>@ %ORACLE_HOME%\rdbms\demo\dmb1prot.sql
SQL>@ %ORACLE_HOME%\rdbms\demo\dmb1coli.sql
```

Exit SQL\*Plus and use the SQL\*Loader utility to load data into the `SwissProt` database in the schema of the data mining user. From the command prompt, change to the `\rdbms\demo` directory under Oracle home and execute the following command.

```
>sqlldr dmuser3/dmuser3_password
        control=dmb1prot.ctl data=dmb1prot.txt log=dmb1prot.log
```

## Running the BLAST Table Functions

The sample program `dmbldemo.sql` contains multiple invocations of BLAST table functions. You can run them all at once by running the `dmbldemo.sql` script, or you can copy individual table functions to the SQL\*Plus command line and execute them individually.

To run the sample program, log in to SQL\*Plus as the data mining user and run the `dmbldemo.sql` script as shown in the following example.

```
SQL>connect dmuser3/dmuser3_password
SQL> @ %ORACLE_HOME%\rdbms\demo\dmbldemo.sql
```

The following example shows how you could execute the sample `BLASTP_MATCH` table function at the SQL\*Plus command line. `BLASTP_MATCH` is run against all the human proteins in the `SwissProt` database.

```
SQL>connect dmuser3/dmuser3_password
SQL>SET serveroutput ON
SQL>SET trimspool ON
SQL>SET pages 10000
SQL>column seq_id format a10
SQL>select T_SEQ_ID AS seq_id, score, EXPECT as evaluate
        from TABLE(
                BLASTP_MATCH (
                        (select sequence from query_db),
                        CURSOR(SELECT seq_id, seq_data
                                FROM swissprot
                                WHERE organism = 'Homo sapiens (Human)'),
                        1,
                        -1,
                        0,
                        0,
                        'BLOSUM62',
                        10,
                        0,
                        0,
                        0,
                        0,
                        0)
        )
```

```
order by score,seq_id;
```

The output of this table function is shown as follows.

| SEQ_ID | SCORE | EVALUE     |
|--------|-------|------------|
| P58107 | 49    | 7.24297332 |
| P31947 | 169   | 8.8130E-14 |
| P27348 | 198   | 3.8228E-17 |
| Q04917 | 198   | 3.8228E-17 |
| P31946 | 205   | 5.8977E-18 |

---

---

# Index

## A

---

accounts  
  unlocking, 1-5  
add-in, 3-1

## B

---

Basic, 4-16  
BLAST  
  sample programs, 4-4, 4-16  
  table functions, 4-17

## C

---

component validation, 1-9  
CREATE DIRECTORY command (SQL), 2-7

## D

---

Data Miner, 3-4  
Data Mining Scoring Engine  
  installation, 1-9  
Data Pump, 2-7  
database  
  access to an existing, 1-1  
Database Configuration Assistant, 2-2  
Database Control, 2-1  
database directories, 2-7  
database installation  
  Linux, 1-8  
  Windows platform, 1-3  
DBMS\_DATA\_MINING package (PL/SQL), 2-8,  
  3-7, 4-10, 4-12  
DBMS\_DATA\_MINING\_TRANSFORM package  
  (PL/SQL), 3-7, 4-10  
DBMS\_PREDICTIVE\_ANALYTICS package  
  (PL/SQL), 4-12  
DM\$P\_MODEL\_EXPIMP\_TEMP table, 2-8  
DM\$P\_MODEL\_TABKEY\_TEMP table, 2-8  
DM\_NESTED\_NUMERICALS, 4-15  
dmsh script, 4-4, 4-6, 4-15  
dmshgrants script, 2-5, 4-4, 4-6  
DMSYS user, 2-6  
documentation  
  Oracle Database, 1-2  
downgrading from release 10.2, 1-12

## E

---

Enterprise Manager, 2-1  
environment variables (Linux)  
  setting, 1-8  
EXP\_FULL\_DATABASE privilege, 2-8  
expdp command, 2-7  
exporting models, 2-6, 4-5, 4-12

## F

---

file system access, 2-7

## G

---

GRANT command (SQL), 2-6, 2-7

## H

---

hardware requirements  
  Windows database installation, 1-2

## I

---

IMP\_FULL\_DATABASE privilege, 2-8  
impdp command, 2-7  
importing models, 2-6, 4-5, 4-12  
initialization files  
  modifying, 1-8  
installation disks  
  list, 1-2  
installer, 1-2

## J

---

Java APIs, 3-7  
  sample programs, 4-4, 4-12  
Java version, 3-5  
javax.datamining.task.ExportTask interface, 2-8  
javax.datamining.task.ImportTask utility, 2-8  
JDeveloper, 3-7  
JDM 1.0 standard, 2-8, 4-12  
JSR-73 specification, 2-8, 4-12

## L

---

Linux  
administration, 2-3  
log files  
export example, 2-9  
import example, 2-10

## N

---

nested tables, 4-15  
net service name, 3-3

## O

---

operating system requirements, 1-2  
Oracle Data Miner, 3-4  
Oracle Data Pump, 2-7  
Oracle Enterprise Manager, 2-1  
Oracle Net Service Name, 3-3  
Oracle Spreadsheet Add-In for Predictive Analytics, 3-1  
Oracle Text, 4-14  
ORACLE\_HOME  
existing, 1-3  
oracle.dmt.jdm.transform.text.OraTextTransform interface, 4-16

## P

---

passwords  
resetting system, 1-5  
PL/SQL APIs, 3-7  
sample programs, 4-3, 4-10

## R

---

RAC installations  
creating tablespaces, 1-8  
runInstaller program, 1-8

## S

---

sample programs, 4-10  
BLAST, 4-4, 4-16  
data, 4-6  
installing, 1-7  
Java, 4-4, 4-12  
listing, 4-2  
obtaining, 4-1  
PL/SQL, 4-3, 4-10  
set up, 4-5  
SQL scoring functions, 4-11  
text mining, 4-14  
sample schemas  
installing, 1-5, 4-2, 4-5  
Scoring Engine  
installation, 1-9  
service name, 3-3  
Spreadsheet Add-In for Predictive Analytics, 3-1

SQL scoring functions, 4-10, 4-11  
SQL\*Plus, 2-2

## T

---

table access, 2-6  
tablespaces, 2-4  
text mining  
java, 4-16  
PL/SQL, 4-15  
sample programs, 4-14

## U

---

Universal Installer, 1-2  
upgrading from release 10.2, 1-10  
URLs  
installation, 1-6  
users  
creating for data mining, 2-5, 4-6

## W

---

Web pages  
installation, 1-6  
Windows administration, 2-1  
Windows database installation  
hardware requirements, 1-2  
operating systems, 1-2