

[Skip Headers](#)

Oracle® Database Upgrade Guide
10g Release 1 (10.1)
Part Number B10763-02

[Home](#) [Book List](#) [Contents](#) [Index](#) [Master Index](#) [Feedback](#)

[View PDF](#)

3 Upgrading a Database to the New Oracle Database 10g Release

This chapter guides you through the process of upgrading a database to the new Oracle Database 10g release. This chapter covers the following topics:

- [System Considerations and Requirements](#)
- [Install the Release 10.1 Oracle Software](#)
- [Upgrade the Database Using the Database Upgrade Assistant](#)
- [Upgrade the Database Manually](#)

See Also:

Some aspects of upgrading are operating system-specific. See your operating system-specific Oracle documentation for additional instructions about upgrading on your operating system.

System Considerations and Requirements

The following sections discuss system considerations and requirements.

Upgrading a Cluster Database

If you are upgrading a cluster database, then most of the actions described in this chapter should be performed on only one node of the system. So, perform the actions described in this chapter on only one node unless instructed otherwise in a particular step.

Gather Optimizer Statistics Before the Upgrade

When upgrading to Oracle Database 10g, optimizer statistics will be collected for dictionary tables that lack statistics. This statistics collection could be time consuming for databases with a large number of dictionary tables, but it will only occur for those tables that lack statistics or are significantly changed during the upgrade.

For databases that are upgraded from Oracle9i, it is possible to decrease the downtime during the upgrade by collecting statistics for the dictionary prior to the upgrade. The following two scripts collect statistics for dictionary objects in Oracle9i.

This process should be tested on a test database just like any other aspect of the upgrade. Also, some schemas referenced in these scripts may not exist if some database components have not been installed.

```
--
-- This script collect stats for system component schemas.
-- The stats collection may give error if a particular component
-- schema does not exist in the database. This can happen
-- if a component is not installed or if it is invalid.
--
-- This script must be run connected AS SYSDBA using SQL*Plus.
--
```

```
spool gdict
```

```
grant analyze any to sys;
```

```
exec dbms_stats.gather_schema_stats('WMSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('MDSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('CTXSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('XDB',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('WKSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('LBACSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('OLAPSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('DMSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('ODM',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('ORDSYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('ORDPLUGINS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('SI_INFORMTN_SCHEMA',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('OUTLN',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('DBSNMP',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('SYSTEM',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
exec dbms_stats.gather_schema_stats('SYS',options=>'GATHER', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);

spool off

--
-- This script creates the stats table, 'dictstattab' and
-- exports the stats for the RDBMS component schemas into it.
-- The export will give error if a particular component
-- schema does not exist in the database. This can happen
-- if a component is not installed or if it is invalid.
--
-- This will be useful incase you want to import the stats back
-- example:
-- Following stmt imports the stats for SYS schema after
-- deleting the existing stats.
-- exec dbms_stats.delete_schema_stats('SYS');
-- exec dbms_stats.import_schema_stats('SYS','dictstattab');
--
-- This script must be run connected AS SYSDBA using SQL*Plus.
--

spool sdic

grant analyze any to sys;

exec dbms_stats.create_stat_table('SYS','dictstattab');

exec dbms_stats.export_schema_stats('WMSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('MDSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('CTXSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('XDB','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('WKSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('LBACSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('OLAPSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('DMSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('ODM','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('ORDSYS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('ORDPLUGINS','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('SI_INFORMTN_SCHEMA','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('OUTLN','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('DBSNMP','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('SYSTEM','dictstattab',statown => 'SYS');
exec dbms_stats.export_schema_stats('SYS','dictstattab',statown => 'SYS');
```

spool off

Upgrading Your Operating System

If required, upgrade your operating system before continuing with your database upgrade.

See Also:

- The Oracle Installation Guide for your platform to determine whether you need to upgrade your operating system
- Your operating system-specific Oracle documentation for information on how to perform an operating system upgrade
-

Migrating Data to a Different Operating System

When using the Database Upgrade Assistant or when performing a manual upgrade, you *cannot* migrate data in a database on one operating system to a database on another operating system. For example, you cannot migrate data in an Oracle9i database on Solaris to an Oracle Database 10g database on Windows 2000 using the Database Upgrade Assistant. However, you normally can use Export/Import to migrate data between databases on different operating systems.

Install the Release 10.1 Oracle Software

Complete the following steps to install the release 10.1 software:

1. If your operating system is UNIX, then make sure you are logged in as a user with write permission to the Oracle home and Oracle base directories, as well as all of their subdirectories.
2. Follow the instructions in your Oracle operating system-specific documentation to prepare for installation and start the Oracle Universal Installer.

If you are upgrading a cluster database, then see [Oracle Real Application Clusters Installation and Configuration Guide](#) for additional installation instructions.

3. At the Welcome screen of the Oracle Universal Installer, click Next. The File Locations screen appears.

If you need help at any screen or want to consult more documentation about the Oracle Universal Installer, then click the Help button to open the online help.

4. At the File Locations screen, complete the following steps:
 - a. Do not change the text in the Source field. This is the location of files for installation.
 - b. Enter the name of a new Oracle home in the Destination Name field.
 - c. Enter the complete path of the Oracle home directory where you want to install the new release in the Destination Path field. Click the Browse button to navigate to the directory.

Note:

You must install the new Oracle Database release in a new Oracle home that is separate from the old release.

- d. Click Next.

The Installation Types screen appears.

5. At the Installation Types screen, complete the following steps:
 - a. Select Enterprise Edition, Standard Edition, or Custom Installation.
 - b. Click Next.

If you chose Enterprise Edition or Standard Edition, then the Database Configuration screen appears.

If you chose Custom Installation, then the Available Product Components screen appears.

6. If the Database Configuration screen appears, then complete the following steps:
 - a. Select Software Only.
 - b. Click Next.

Note:

Normally, you should not install a starter database if you are upgrading an existing database.

7. If the Available Product Components screen appears, then complete the following steps:
 - a. Choose the product components you want to install.
 - b. Click Next.

Make sure you install all of the options you installed with the previous database, assuming you do not want to discontinue use of a particular option. For example, if you installed Oracle Text in the previous database, then you should install Oracle Text in the new Oracle Database.

8. If you are installing Real Application Clusters, then, at the Cluster Node Selection screen, select the nodes onto which you want the software installed. Then, click Next.
9. At the Create Database screen, select the No option, indicating that you do not want to create a database because you are upgrading an existing database. Then, click Next.
10. Respond to the remaining screens that enable you to specify your custom installation settings, until you reach the Upgrading an Existing Database screen.
11. At the Upgrading an Existing Database screen, complete the following steps:
 - a. To upgrade a database using the Database Upgrade Assistant, select the Upgrade an Existing Database check box.

To upgrade a database manually, or to start the Database Upgrade Assistant independently after installation is complete, do not select the Upgrade an Existing Database check box.
 - b. Click Next.
12. At the Summary screen, make sure all of the settings and choices are correct for your installation. Then, click Install. The Oracle Universal Installer performs the installation.

When installation is complete, one or more assistants may be started. If you chose to run the Database Upgrade Assistant during installation, then you are ready to proceed with the upgrade when the Database Upgrade Assistant is started. See "[Upgrade the Database](#)".

When installation has completed successfully, click the Exit button to close the Oracle Universal Installer.

Upgrade the Database Using the Database Upgrade Assistant

The following sections guide you through the process of upgrading a database using the Database Upgrade Assistant:

- [Starting the Database Upgrade Assistant](#)
- [Database Upgrade Assistant Command Line Options](#)
- [Upgrade the Database](#)
- [Using the Database Upgrade Assistant in Silent Mode](#)

Starting the Database Upgrade Assistant

If you installed the new Oracle Database 10g release and specified that you are upgrading an existing database, then the Database Upgrade Assistant is started automatically. See "[Upgrade the Database](#)". However, if you did not specify that you are upgrading an existing database, then you can start the Database Upgrade Assistant independently after installation is complete.

Complete the following steps to start the Database Upgrade Assistant:

1. In the environment of the new Oracle Database 10g release, start the Database Upgrade Assistant.

On UNIX platforms, enter the following command at a system prompt:

```
dbua
```

Note:

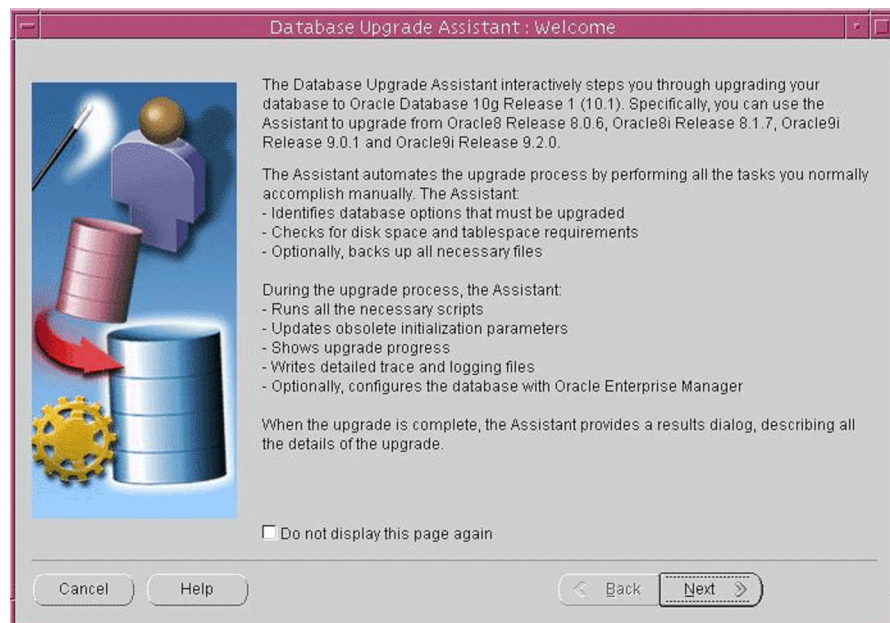
The dbua executable is usually located in `ORACLE_HOME/bin`.

On Windows operating systems, choose:

```
Start > Programs > Oracle - HOME_NAME > Configuration and Migration Tools >
Database Upgrade Assistant
```

When the Database Upgrade Assistant starts, its Welcome screen appears. [Figure 3-1](#) shows the Welcome screen of the Database Upgrade Assistant.

Figure 3-1 Welcome Screen of the Database Upgrade Assistant



[Description of the illustration dbua.gif](#)

Database Upgrade Assistant Command Line Options

The Database Upgrade Assistant supports several command line options. You can specify all valid options from the command line using the following syntax:

```
dbua [ -silent ] [ -dbName SID ]
[ -disableUpgradeScriptLogging ] [ -backupLocation directory ]
[ -postUpgradeScripts script [ , script ] ... ]
[ -initParam parameter=value [ , parameter=value ] ... ]
[ -emConfiguration { LOCAL | CENTRAL | NOBACKUP | NOEMAIL | NONE }
-dbsnmpPassword password -sysmanPassword password
[ -hostUserName hostname -hostUserPassword password
```

```
-backupSchedule hh:mm
]
[ -smtpServer server_name -emailAddress address ]
-centralAgent location
]
[ -recoveryAreaDestination directory ] [ -h | -help ]
```

[Table 3-1](#) describes the various options and their parameters that are supported by the Database Upgrade Assistant.

Table 3-1 Database Upgrade Assistant Command Line options

Option	Description
-silent	Specifies that the Database Upgrade Assistant should operate in silent mode. See "Using the Database Upgrade Assistant in Silent Mode" .
-dbName SID	Specifies the system identifier (SID) of the database to upgrade
-disableUpgradeScriptLogging	This option disables the detailed log generation for running SQL scripts during the upgrade process. This is enabled by default. To enable the log generation, do not specify this option.
-backupLocation directory	Specifies a directory to back up your database before the upgrade starts
-postUpgradeScripts script [, script] ...	Specifies a comma-separated list of SQL scripts. Specify complete pathnames. The scripts will be executed at the end of the upgrade.
-initParam parameter=value [, parameter=value] ...	Specifies a comma-separated list of initialization parameter values of the form <i>name=value</i>
-emConfiguration { LOCAL CENTRAL NOBACKUP NOEMAIL NONE }	Specifies Enterprise Manager management options: <ul style="list-style-type: none"> • LOCAL - Database is locally managed by Enterprise Manager • CENTRAL - Database is centrally managed by Enterprise Manager • NOBACKUP - Automatic daily backups of the database are not enabled • NOEMAIL - E-mail notifications are not enabled • NONE - Database is not managed by Enterprise Manager
-dbsnmpPassword password	Specifies the DBSNMP user password
-sysmanPassword password	Specifies the SYSMAN user password
-hostUserName hostname	Specifies the host user name for the Enterprise Manager backup job
-hostUserPassword password	Specifies the host user password for the Enterprise Manager backup job
-backupSchedule hh:mm	Specifies the daily backup schedule in the form <i>hh:mm</i> (hours and minutes)
-smtpServer server_name	Specifies the outgoing mail (SMTP) server for E-mail notifications
-emailAddress address	Specifies the E-mail address for E-mail notifications
-centralAgent location	Specifies the Enterprise Manager central agent location
-recoveryAreaDestination directory	Specifies the destination directory for all recovery files
-h -help	Displays usage help for the Database Upgrade Assistant

For example, the following command selects a database named ORCL for an upgrade:

```
dbua -dbName ORCL
```

Upgrade the Database

When the Welcome screen of the Database Upgrade Assistant appears ([Figure 3-1](#)), you are ready to proceed with the upgrade. Complete the following steps to upgrade the database:

1. At the Welcome screen of the Database Upgrade Assistant, make sure the database being upgraded meets the specified conditions. Then, click Next.

If you need help at any screen or want to consult more documentation about the Database Upgrade Assistant, then click the Help button to open the online help.

2. At the Selecting a Database Instance screen, select the database you want to upgrade from the Available Databases table. Then, click Next.

You can select only one database at a time. If you are running the Database Upgrade Assistant from a user account that does not have `SYSDBA` privileges, then you must enter the user name and password credentials to enable `SYSDBA` privileges for the selected database.

The database you select must already be started. The Database Upgrade Assistant analyzes the database, performing pre-upgrade checks and displaying warnings as necessary:

- It checks for any redo log files whose size is less than 4 MB. If such files are found, then the Database Upgrade Assistant gives the option to drop/create new redo log files.
- It checks the parameter file for any obsolete or deprecated initialization parameters

3. At the Creating the SYSAUX Tablespace screen, specify the attributes for the `SYSAUX` tablespace, which is added automatically to all new Oracle Database 10g databases you create. Then, click Next.

See Also:

[Oracle Database Administrator's Guide](#) for more information about the `SYSAUX` tablespace

Many of the attributes of the `SYSAUX` tablespace are set automatically and cannot be modified. For example, the `SYSAUX` tablespace is set to use Automatic Segment-Space Management. However, you can specify the location of the datafile, the default size of the `SYSAUX` tablespace, and its autoextend attributes.

Note:

If you specify an existing datafile for the `SYSAUX` tablespace, then you must select Reuse Existing File Name. Otherwise, the Database Upgrade Assistant alerts you to the fact that the file already exists.

4. At the Recompiling Invalid Objects screen, decide whether you want the Database Upgrade Assistant to recompile all invalid PL/SQL modules after the upgrade is complete. Then, click Next.

When you upgrade a database to the new Oracle Database 10g release, many of the PL/SQL modules in your database will become invalid. As a result, all existing PL/SQL modules in an `INVALID` state must be recompiled, such as packages, procedures, types, and so on.

By default, the Oracle Database recompiles invalid PL/SQL modules as they are used. For example, if an invalid PL/SQL module is called, it will first be recompiled before it is actually executed. The time it takes to recompile the module can result in poor performance as you begin to use your newly upgraded database.

To eliminate these performance issues, select Recompile invalid objects at the end of upgrade. When you select this option, the Database Upgrade Assistant recompiles all the invalid PL/SQL modules immediately after the upgrade is performed. This will ensure that you will not experience any performance issues later, as you begin using your newly upgraded database.

Note:

Selecting Recompile invalid objects at the end of upgrade is equivalent to running the `ORACLE_HOME/rdbms/admin/utlrp.sql` script, which is used to recompile stored PL/SQL and Java code.

The task of recompiling all the invalid PL/SQL modules in your database can take a significant amount of time and increase the time it takes to complete your database upgrade. If you have multiple CPUs, then you can reduce the time it takes to perform this task by taking advantage of parallel processing on your available CPUs. If you have multiple CPUs available, then the Database Upgrade Assistant automatically adds an additional section to the Recompile Invalid Objects screen and automatically determines the number of CPUs you have available.

The Database Upgrade Assistant also provides a recommended degree of parallelism, which determines how many parallel processes are used to recompile your invalid PL/SQL modules. Specifically, the Database Upgrade Assistant sets the degree of parallelism to one less than the number of CPUs you have available. For example, if you have three CPUs available for processing, then the Database Upgrade Assistant selects 2 from the Degree of Parallelism menu. You can adjust this default value by selecting a new value from the Degree of Parallelism menu.

5. At the Choosing a Database Backup Procedure screen, specify whether or not you want the Database Upgrade Assistant to back up your database for you. Then, click Next. If you choose not to use the Database Upgrade Assistant for your backup, then Oracle assumes you have already backed up your database using your own backup procedures.

Note:

Oracle strongly recommends that you back up your database before the upgrade. If errors occur during the upgrade, then you may need to restore the database from the backup.

If you use the Database Upgrade Assistant to back up your database, then the Database Upgrade Assistant will make a copy of all your database files in the directory you specify in the Backup Directory field. The Database Upgrade Assistant will perform this cold backup automatically after it shuts down the database and before it begins performing the upgrade procedure. The cold backup will not compress your database files and the backup directory must be a valid file system path. You cannot specify a raw device for the cold backup files.

In addition, the Database Upgrade Assistant creates a batch file in the specified directory. You can use this batch file to restore the database files:

- On Windows operating systems, the file is called `db_name_restore.bat`.
- On UNIX platforms, the file is called `db_name_restore.sh`.

6. At the Management Options screen, you have the option of setting up your database so it can be managed with Enterprise Manager. Enterprise Manager provides Web-based management tools for managing individual database instances, as well as central management tools for managing your entire Oracle environment, including multiple databases, hosts, application servers, and other components of your network.
 - a. When you run the Database Upgrade Assistant, the assistant checks to see if the Oracle Management Agent has been installed on the host computer. If the assistant locates an Oracle Management Agent, select the Grid Control option and select an Oracle Management Service from the drop-down list. When you finish installing the Oracle Database, the database will automatically be available as a managed target within the Oracle Enterprise Manager Grid Control.
 - b. If you are not centrally managing your Oracle environment, you can still use Enterprise Manager to manage your database. When you install an Oracle Database, you automatically install the Oracle Enterprise Manager Database Control, which provides Web-based features for monitoring and administering the single-instance or cluster database you are installing.

To configure the database so it can be managed with the Oracle Enterprise Manager Database Control, select the Database Control option.

- c. When you select the Database Control management option, you can configure Enterprise Manager so that E-mail notifications will be enabled immediately upon installation.

Select Enable Email Notifications if you want the SYSMAN user (the default Super Administrator and owner of the Management Repository schema) to receive email notification when a metric for a specified condition reaches a critical or warning threshold. For example, Enterprise Manager can send an email when a target goes down or when there are database space usage problems.

- d. If you select the Database Control management option, you can also enable automatic daily backups of your entire database.

Select Enable Daily Backups to use the Oracle-suggested backup strategy to back up your entire database with a minimum amount of configuration. Later, you can use Enterprise Manager to customize your backup strategy further.

When you select this option, Enterprise Manager will be configured to back up your database, based on the scheduled start time you enter on this page, immediately after you finish installing the Oracle Database. Enterprise Manager will back up the database to the Flash Recovery Area that you specify later on the Recovery Configuration screen of the Database Upgrade Assistant.

- e. After you have made your choices, click **Next**.

7. At the Database Credentials screen, secure your database with passwords for the Enterprise Manager accounts. You can set a single password, which will be applied to each of the listed Enterprise Manager user accounts, or enhance the security of the accounts by providing unique passwords for each user.
8. At the Recovery Configuration screen, specify a flash recovery area and enable archiving. When you are managing your database, it is important to configure the database so you can recover your data in the event of a system failure.

The Flash Recovery Area can be used to recover data that would otherwise be lost during a system failure; this location is also used by Enterprise Manager if you have enabled local management and daily backups on the Management Options screen shown previously in the Database Upgrade Assistant.

9. At the Network Configuration for the database screen, you have two tabs:

The Listeners tab is displayed if you have more than one listener in the release 10.1 Oracle home. Select the listeners in the release 10.1 Oracle home for which you would like to register the upgraded database.

The Directory Service tab shows up if you have directory services configured in the release 10.1 Oracle home. You can select to either register or not register the upgraded database with the directory service.

10. At the Summary screen, make sure all of the specifications are correct. If anything is incorrect, then click Back until you can correct the specification. If everything is correct, then click Finish.

The Database Upgrade Assistant lists the initialization parameters that will be set for the database during the upgrade. The `COMPATIBLE` initialization parameter will be set to at least 9.2.0.

See Also:

[Chapter 5, "Compatibility and Interoperability"](#) for information about setting the `COMPATIBLE` initialization parameter after the upgrade

11. A Progress dialog appears and the Database Upgrade Assistant begins to perform the upgrade.

You may encounter error messages with Ignore and Abort choices. If other errors appear, then you must address them accordingly. If an error is severe and cannot be handled during the upgrade, then you have the following choices:

- If Ignore is presented as a choice in the message, then clicking the button will ignore the error and proceed with the upgrade. The errors ignored are logged and shown later in the summary.

This causes the Database Upgrade Assistant to display the step as skipped and move on to the next step in the upgrade, ignoring this and any dependent steps. After the upgrade is complete, you can fix the problem, restart the Database Upgrade Assistant, and complete the skipped steps.

- If Ignore is not presented as a choice in the message, then you need to abort the process by clicking the Abort button.

This will abort the upgrade process. The Database Upgrade Assistant prompts you to restore the database if the database backup was taken by the Database Upgrade Assistant.

After the database has been restored, you need to correct the cause of the error and restart the Database Upgrade Assistant to perform the upgrade again.

If you do not want to restore the database, then the Database Upgrade Assistant leaves the database in its present state so that you can proceed with a manual upgrade.

After the upgrade has completed, the following message is displayed in the Progress dialog:

Upgrade has been completed. Click the "OK" button to see the results of the upgrade.

Click the OK button.

12. At the Checking Upgrade Results screen, you can examine the results of the upgrade, manage the passwords in the upgraded database, and, if necessary, restore the original database settings.

The upgrade results summary includes a description of the original and upgraded databases and changes made to the initialization parameters. The upgrade results also include a Step Execution Summary that describes the steps performed during the database upgrade. For each step in the process, the summary provides the step name, the log file for the step, and the status. In some cases, you can click the status to display details about the execution step. The Step Execution Summary also includes the directory where the various log files are stored after the upgrade. You can examine any of these log files to obtain more details about the upgrade process.

Note:

An HTML version of the Upgrade Results is also saved in the log files directory.

The Password Management section of the screen allows you to unlock and set passwords for various users in the newly upgraded database. Click Configure Database Passwords to display the Password Management dialog box. The Password Management dialog box allows you to change the default password for a user after you upgrade the database. For security reasons, all users are locked except for the following users:

- SYS
- SYSTEM

If you have enabled Local Management with Enterprise Manager, then the `SYSMAN` and `DBSNMP` accounts are also unlocked. These accounts provide Enterprise Manager with access to the database so it can gather monitoring data and so you can perform administration tasks with Enterprise Manager.

If you have enabled Central Management with Enterprise Manager, then the `DBSNMP` account is unlocked, as well as the `SYS` and `SYSTEM` user accounts.

Note:

To prevent unauthorized use of the database, Oracle recommends that you change all user passwords immediately after you upgrade your database.

If you are not satisfied with the upgrade results, then click Restore. Depending on the method you used to back up your database, the Restore operation performs one of two tasks:

- If you used the Database Upgrade Assistant to back up your database, then clicking Restore will restore the original database and the original database settings from the backup.
- If you used your own backup procedure to back up the database, then clicking Restore will restore only the original database settings. To restore the database itself, you must restore the backup you created with your own backup utilities.

If you are satisfied with the upgrade results, then click Exit to quit the Database Upgrade Assistant and use your newly upgraded database. The Database Upgrade Assistant removes the entry of the upgraded database from the old `listener.ora` file and reloads the listener of the old database.

- a. The Database Upgrade Assistant modifies the `SID_DESC` entry for the upgraded database in the Oracle Database `listener.ora` file in one of the following ways:

A simple case: Suppose the old `listener.ora` has the following `SID_DESC` entry:

```
...
(SID_DESC =
  (SID_NAME = ORCL)
)
...
```

If the database name is `SAL`, the domain name is `COM`, and the Oracle home is `/oracle/product/9.2`, then the assistant adds the following entry:

```
...
(SID_DESC =
  (GLOBAL_DBNAME = sal.com)
  (ORACLE_HOME = /oracle/product/9.2)
  (SID_NAME = SAL)
)
...
```

A more complicated case: Suppose the old `listener.ora` has the following `SID_DESC` entry:

```
...
(SID_DESC =
  (GLOBAL_DBNAME = an_entry)
  (SID_NAME = ORCL)
)
...
```

If `an_entry` does not match the `GLOBAL_DBNAME` of the migrated database, and if the database name is `SAL`, the domain name is `COM`, and the Oracle home is `/oracle/product/9.2`, then the assistant adds the following entry:

```
...
(SID_DESC =
  (GLOBAL_DBNAME = sal.com)
  (ORACLE_HOME = /oracle/product/9.2)
  (SID_NAME = SAL)
)
...
```

This entry is the same as the entry in the simple case, but the Database Upgrade Assistant also adds the entry `an_entry` to the `SERVICE_NAMES` parameter in the `listener.ora` file. Therefore, the Database Upgrade Assistant changes the `SERVICE_NAMES` parameter to the following:

```
SERVICE_NAMES = sal.com, an_entry
```

- b. The Database Upgrade Assistant removes the entry of the upgraded database from the old `listener.ora` file.
 - c. The Database Upgrade Assistant reloads the `listener.ora` file in both the old and new Oracle Database environments.
13. At the **Changes in Default Behavior** screen, the Database Upgrade Assistant displays some changes in behavior of Oracle Database 10g from that of previous releases. In some cases the default values of some initialization parameters have changed. In other cases some new behavior/requirement has been introduced that may affect current scripts or applications.
 14. Complete the procedures described in [Chapter 4, "After Upgrading a Database"](#).

WARNING:

If you retain the old Oracle software, then never start the upgraded database with the old Oracle software. Only start the database with the executables in the new Oracle Database installation. Also, before you remove the old Oracle environment, make sure you relocate any datafiles in that environment to the new Oracle Database environment. See the [Oracle Database Administrator's Guide](#) for information about relocating datafiles.

Using the Database Upgrade Assistant in Silent Mode

When invoked with the `-silent` command line option, the Database Upgrade Assistant operates in silent mode. In silent mode, the Database Upgrade Assistant does not present a user interface. It also writes any messages (including information, errors, and warnings) to a log file.

For example, the following command upgrades a database named `ORCL` in silent mode:

```
dbua -silent -dbName ORCL
```

Upgrade the Database Manually

The following sections guide you through the process of performing a manual upgrade:

- [Analyze the Database to be Upgraded](#)
- [Backup the Database](#)
- [Upgrade the Database](#)
- [Troubleshooting the Upgrade](#)
- [Abandoning the Upgrade](#)

Analyze the Database to be Upgraded

Before you upgrade your database to the new Oracle Database 10g release, you must run the Pre-Upgrade Information Tool. The Upgrade Information Tool is a SQL script that ships with the new Oracle Database 10g release, and must be copied to and run from the environment of the database being upgraded. Complete the following steps to run the Upgrade Information Tool:

1. Log in to the system as the owner of the Oracle home directory of the new Oracle Database 10g release.
2. Copy the following file from the `ORACLE_HOME/rdbms/admin` directory of the new Oracle Database 10g release to a directory outside of the Oracle home, such as the temporary directory on your system:

- `utlu101i.sql`

Make a note of the new location of this file.

3. Log in to the system as the owner of the Oracle home directory of the database being upgraded.
4. Change to the directory outside of the Oracle home directory that you copied files to in Step 2.
5. Start `SQL*Plus`.
6. Connect to the database instance as a user with `SYSDBA` privileges.
7. Set the system to spool results to a log file for later analysis:

```
SQL> SPOOL info.log
```

8. Run `utlu101i.sql`:

```
SQL> @utlu101i.sql
```

9. Turn off the spooling of script results to the log file:

```
SQL> SPOOL OFF
```

Then, check the spool file and examine the output of the upgrade information tool. You named the spool file in Step 7; the suggested name was `info.log`.

The following example displays the output of the Upgrade Information Tool for a release 9.0.1.0.0 database named `TEST`.

```
Oracle Database 10.1 Upgrade Information Tool      MM-DD-YYYY HH:MM:SS
'
*****
```

```
Database:
-----
--> name: TEST
--> version: 9.0.1.0.0
--> compatibility: 9.0.0
WARNING: Database compatibility must be set to 9.2.0 prior to upgrade.
.
*****
Logfiles: [make adjustments in the current environment]
-----
-- The existing log files are adequate. No changes are required.
.
*****
Tablespaces: [make adjustments in the current environment]
-----
--> SYSTEM tablespace is adequate for the upgrade.
.... owner: SYS
.... minimum required size: 466 MB
--> CWWMLITE tablespace is adequate for the upgrade.
.... owner: OLAPSYS
.... minimum required size: 13 MB
--> DRSYS tablespace is adequate for the upgrade.
.... owner: CTXSYS
.... minimum required size: 8 MB
.
*****
Options: [present in existing database]
-----
--> Partitioning
--> Spatial
WARNING: Listed option(s) must be installed with Oracle Database 10.1
.
*****
Update Parameters: [Update Oracle Database 10.1 init.ora or spfile]
-----
WARNING: --> "shared_pool_size" needs to be increased to at least "100663296"
WARNING: --> "pga_aggregate_target" needs to be increased to at least "25165824"
WARNING: --> "large_pool_size" needs to be increased to at least "8388608"
--> "java_pool_size" is already at "67108864" calculated new value is "67108864"
.
*****
Deprecated Parameters: [Update Oracle Database 10.1 init.ora or spfile]
-----
-- No deprecated parameters found. No changes are required.
.
*****
Obsolete Parameters: [Update Oracle Database 10.1 init.ora or spfile]
-----
--> "undo_suppress_errors"
.
*****
Components: [The following database components will be upgraded or installed]
-----
--> Oracle Catalog Views          [upgrade]
--> Oracle Packages and Types     [upgrade]
--> JServer JAVA Virtual Machine  [upgrade]
--> Oracle XDK for Java           [upgrade]
--> Oracle Java Packages          [install]
--> Oracle Workspace Manager      [upgrade]
--> OLAP Catalog                  [upgrade]
--> Oracle interMedia             [upgrade]
--> Spatial                       [upgrade]
--> Oracle Text                   [upgrade]
--> Oracle Ultra Search           [upgrade]
.
*****
.
*****
SYSaux Tablespace: [Create tablespace in Oracle Database 10.1 environment]
-----
--> New "SYSaux" tablespace
.... minimum required size for database upgrade: 500 MB
Please create the new SYSaux Tablespace AFTER the Oracle Database
10.1 server is started and BEFORE you invoke the upgrade script.
.
*****
```

The sections which follow describe the output of the Upgrade Information Tool.

Database

This section displays global database information about the current database, such as the database name, release number, and compatibility level. A warning is displayed if the `COMPATIBLE` initialization parameter needs to be adjusted before the database is upgraded.

Logfiles

This section displays a list of redo log files in the current database whose size is less than 4 MB. For each log file, the file name, group number, and recommended size is displayed. New files of at least 4 MB (preferably 10 MB) need to be created in the current database. Any redo log files less than 4 MB must be dropped before the database is upgraded.

Tablespaces

This section displays a list of tablespaces in the current database. For each tablespace, the tablespace name, owner, and minimum required size is displayed. In addition, a message is displayed if the tablespace is adequate for the upgrade. If the tablespace does not have enough free space, then space must be added to the tablespace in the current database. Tablespace adjustments need to be made before the database is upgraded.

Options

This section displays a list of options in the current database that must be available in the new Oracle Database 10g release before the database is upgraded.

Update Parameters

This section displays a list of initialization parameters in the parameter file of the current database that must be adjusted before the database is upgraded. The adjustments need to be made to the parameter file after it is copied to the new Oracle Database 10g release.

See Also:

[Appendix A, "Initialization Parameter and Data Dictionary Changes"](#) for more information about changes to initialization parameters in the new Oracle Database 10g release

Deprecated Parameters

This section displays a list of initialization parameters in the parameter file of the current database that are deprecated in the new Oracle Database 10g release.

See Also:

["Deprecated Initialization Parameters"](#) for a list of initialization parameters that are deprecated in the new Oracle Database 10g release

Obsolete Parameters

This section displays a list of initialization parameters in the parameter file of the current database that are obsolete in the new Oracle Database 10g release. Obsolete initialization parameters need to be removed from the parameter file before the database is upgraded.

See Also:

["Obsolete Initialization Parameters"](#) for a list of initialization parameters that are obsolete in the new Oracle Database 10g release

Components

This section displays a list of database components in the new Oracle Database 10g release that will be upgraded or installed when the current database is upgraded.

SYSAUX Tablespace

This section displays the minimum required size for the `SYSAUX` tablespace, which is required in Oracle Database 10g. The `SYSAUX` tablespace must be created after the new Oracle Database 10g release is started and BEFORE the upgrade scripts are invoked.

Backup the Database

After cleanly shutting down the instance following the analysis of the database, you should perform a full backup of the database. Complete the following steps:

1. Sign on to RMAN:

```
rman "target / nocatalog"
```

2. Issue the following RMAN commands:

```
RUN
{
  ALLOCATE CHANNEL chan_name TYPE DISK;
  BACKUP DATABASE FORMAT 'some_backup_directory%U' TAG before_upgrade;
  BACKUP CURRENT CONTROLFILE TO 'save_controlfile_location';
}
```

Caution:

If you encounter any problems with the upgrade, or later wish to abandon the upgrade, then you will need to restore the database from this backup. Therefore, make sure you back up your database now as a precaution.

See Also:

[Oracle Database Backup and Recovery Basics](#) for more information about backing up a database

Upgrade the Database

After analyzing and backing up the database to be upgraded, you are ready to proceed with the manual upgrade. Complete the following steps to upgrade the database:

1. Copy configuration files from the Oracle home of the database being upgraded to the new Oracle Database 10g Oracle home:
 - a. If your parameter file resides within the old environment's Oracle home, then copy it to the new Oracle home. By default, Oracle looks for the parameter file in `ORACLE_HOME/dbs` on UNIX platforms and in `ORACLE_HOME\database` on Windows operating systems. The parameter file can reside anywhere you wish, but it should not reside in the old environment's Oracle home after you upgrade to Oracle Database 10g.
 - b. If your parameter file is a text-based initialization parameter file with either an `IFILE` (include file) or a `SPFILE` (server parameter file) entry, and the file specified in the `IFILE` or `SPFILE` entry resides within the old environment's Oracle home, then copy the file specified by the `IFILE` or `SPFILE` entry to the new Oracle home. The file specified in the `IFILE` or `SPFILE` entry contains additional initialization parameters.
 - c. If you have a password file that resides within the old environment's Oracle home, then move or copy the password file to the new Oracle Database 10g Oracle home.

The name and location of the password file are operating system-specific. On UNIX platforms, the default password file is `ORACLE_HOME/dbs/orapwsid`. On Windows operating systems, the default password file is `ORACLE_HOME\database\pwsid.ora`. In both cases, `sid` is your Oracle instance ID.

- d. If you are upgrading a cluster database and your `initdb_name.ora` file resides within the old environment's Oracle home, then move or copy the `initdb_name.ora` file to the new Oracle home.

Note:

If you are upgrading a cluster database, then perform this step on all nodes in which this cluster database has instances configured.

2. Adjust your parameter file in the new Oracle Database 10g release by completing the following steps:

- a. Remove obsolete initialization parameters and adjust deprecated initialization parameters. Certain parameters are obsolete in the new Oracle Database 10g release, while other parameters have become deprecated. Remove all obsolete parameters from any parameter file that will start a release 10.1 instance. Obsolete parameters may cause errors in the new Oracle Database 10g release. Also, alter any parameter whose syntax has changed in the new Oracle Database 10g release.

The Upgrade Information Tool displays any deprecated parameters and obsolete parameters it finds in the **Deprecated Parameters** and **Obsolete Parameters** sections, respectively.

See Also:

[Appendix A, "Initialization Parameter and Data Dictionary Changes"](#) for a list of initialization parameters that have been deprecated or have become obsolete

- b. Make sure the `COMPATIBLE` initialization parameter is properly set for the new Oracle Database 10g release. The Upgrade Information Tool displays a warning in the **Database** section if `COMPATIBLE` is not properly set.
- c. Make sure the `SHARED_POOL_SIZE` initialization parameter is set to at least 96 MB (for 32-bit platforms) or to at least 144 MB (for 64-bit platforms).
- d. Make sure the `PGA_AGGREGATE_TARGET` initialization parameter is set to at least 24 MB.
- e. Make sure the `LARGE_POOL_SIZE` initialization parameter is set to at least 8 MB.
- f. Make sure the `JAVA_POOL_SIZE` initialization parameter is set to at least 48 MB.
- g. Make sure the `DB_DOMAIN` initialization parameter is set properly.

See Also:

["The DB_DOMAIN Parameter"](#) for more information about setting this initialization parameter

- h. On Windows operating systems, change the `BACKGROUND_DUMP_DEST` and `USER_DUMP_DEST` initialization parameters that point to `RDBMS80` or any other environment variable to point to the following directories instead:

Initialization Parameter	Change Setting To
<code>BACKGROUND_DUMP_DEST</code>	<code>ORACLE_BASE\oradata\DB_NAME</code>
<code>USER_DUMP_DEST</code>	<code>ORACLE_BASE\oradata\DB_NAME\archive</code>

In the settings, substitute the complete Oracle base path for `ORACLE_BASE` and substitute the database name for `DB_NAME`.

- i. Make sure all path names in the parameter file are fully specified. You should not have relative path names in the parameter file.
- j. If the parameter file contains an `IFILE` entry, then change the `IFILE` entry in the parameter file to point to the new location of the include file that you specified in Step [1](#), [b](#). Then, edit the file specified in the `IFILE` entry in the same way that you edited the parameter file in Step [c](#) through Step [i](#).
- k. If you are upgrading a cluster database, then modify the `initdb_name.ora` file in the same way that you modified the parameter file.

Make sure you save all of the files you modified after making these adjustments.

Note:

If you are upgrading a cluster database, then perform this step on all nodes in which this cluster database has instances configured.

3. If you are upgrading a cluster database, then set the `CLUSTER_DATABASE` initialization parameter to `false`. After the upgrade, you must set this initialization parameter back to `true`.
4. Shut down the instance:

```
SQL> SHUTDOWN IMMEDIATE
```

If your operating system is Windows, then complete the following steps:

- a. Stop the OracleService*SID* Oracle service of the database you are upgrading, where *SID* is the instance name. For example, if your *SID* is ORCL, then enter the following at a command prompt:

```
C:\> NET STOP OracleServiceORCL
```

- b. Delete the Oracle service at a command prompt using ORADIM. The following table lists the command to run for each Oracle Database release:

Oracle Database Release	Enter at a Command Prompt
8.0	C:\> ORADIM80 -DELETE -SID <i>SID</i>
8.1 and higher	C:\> ORADIM -DELETE -SID <i>SID</i>

For example, if your Oracle Database release is release 8.0.6 and your *SID* is ORCL, then enter the following command:

```
C:\> ORADIM80 -DELETE -SID ORCL
```

If your Oracle Database release is release 8.1.7 and your *SID* is ORCL, then enter the following command:

```
C:\> ORADIM -DELETE -SID ORCL
```

- c. Create the new Oracle Database 10g service at a command prompt using the ORADIM command of the new Oracle Database release:

```
C:\> ORADIM -NEW -SID SID -INTPWD PASSWORD -MAXUSERS USERS
        -STARTMODE AUTO -PFILE ORACLE_HOME\DATABASE\INITSID.ORA
```

This syntax includes the following variables:

Variable	Description
<i>SID</i>	The same <i>SID</i> name as the <i>SID</i> of the database you are upgrading.
<i>PASSWORD</i>	The password for the new release 10.1 database instance. This is the password for the user connected with SYSDBA privileges. The -INTPWD option is not required. If you do not specify it, then operating system authentication is used, and no password is required.
<i>USERS</i>	The maximum number of users who can be granted SYSDBA and SYSOPER privileges.
<i>ORACLE_HOME</i>	The release 10.1 Oracle home directory. Ensure that you specify the full pathname with the -PFILE option, including drive letter of the Oracle home directory.

For example, if your *SID* is ORCL, your *PASSWORD* is TWxy579, the maximum number of *USERS* is 10, and the *ORACLE_HOME* directory is C:\ORA92, then enter the following command:

```
C:\> ORADIM -NEW -SID ORCL -INTPWD TWxy579 -MAXUSERS 10
        -STARTMODE AUTO -PFILE C:\ORA92\DATABASE\INITORCL.ORA
```

5. If your operating system is UNIX, then make sure that the following environment variables point to the new release 10.1 directories:

- ORACLE_HOME
- PATH
- ORA_NLS10
- LD_LIBRARY_PATH

Note:

If you are upgrading a cluster database, then perform this step on all nodes in which this cluster database has instances configured.

See Also:

Your operating system-specific Oracle Database installation documents for information about setting other important environment variables on your operating system.

6. Log in to the system as the owner of the Oracle home directory of the new Oracle Database 10g release.
7. At a system prompt, change to the `ORACLE_HOME/rdbms/admin` directory.
8. Start SQL*Plus.
9. Connect to the database instance as a user with SYSDBA privileges.
10. Start up the instance by issuing the following command:

```
SQL> STARTUP UPGRADE
```

You may need to use the `PFILE` option to specify the location of your initialization parameter file.

The following are common errors that may occur when attempting to start up the new Oracle Database 10g release.

- If the `COMPATIBLE` initialization parameter is set below 9.2.0:
ORA-00401: the value for parameter compatible is not supported by this release
- If the `CLUSTER_DATABASE` initialization parameter is set to true instead of false:
ORA-39701: database must be mounted EXCLUSIVE for UPGRADE or DOWNGRADE
- If the `STARTUP` command was issued without the `UPGRADE` keyword:
ORA-39700: database must be opened with UPGRADE option
- If a redo log's size is less than 4 MB:
ORA-00336: log file size xxxx blocks is less than minimum 8192 blocks

If errors appear listing obsolete initialization parameters, then make a note of the obsolete initialization parameters and continue with the upgrade. Then, remove the obsolete initialization parameters the next time you shut down the database.

11. Create a SYSAUX tablespace. In Oracle Database 10g, the SYSAUX tablespace is used to consolidate data from a number of tablespaces that were separate in previous releases.

The SYSAUX tablespace must be created with the following mandatory attributes:

- ONLINE
- PERMANENT
- READ WRITE
- EXTENT MANAGEMENT LOCAL
- SEGMENT SPACE MANAGEMENT AUTO

The Upgrade Information Tool provides an estimate of the minimum required size for the SYSAUX tablespace in the **SYSAUX Tablespace** section. [Table 3-2](#) can be used to determine an optimal size for the SYSAUX tablespace.

Table 3-2 Guidelines for sizing the SYSAUX tablespace

Factor	Small	Medium	Large
Number of CPUs	2	8	32
Number of concurrently active sessions	5	20	100
Estimated SYSAUX size at steady state with default config	500 MB	2 GB	5 GB

Factor	Small	Medium	Large
Number of user objects (tables and indexes)	500	5,000	50,000
Estimated SYSAUX size at steady state with default config	500 MB	2 GB	5 GB

The following SQL statement would create a 500 MB SYSAUX tablespace for the database:

```
SQL> CREATE TABLESPACE sysaux DATAFILE 'sysaux01.dbf'
      SIZE 500M REUSE
      EXTENT MANAGEMENT LOCAL
      SEGMENT SPACE MANAGEMENT AUTO
      ONLINE;
```

See Also:

[Oracle Database Administrator's Guide](#) for more information about the SYSAUX tablespace

12. Set the system to spool results to a log file for later verification of success:

```
SQL> SPOOL upgrade.log
```

13. Run `uold_release.sql`, where `old_release` refers to the release you had installed prior to upgrading. See [Table 3-3](#) to choose the correct script. Each script provides a direct upgrade from the release specified in the "Old Release" column. The "Old Release" is the release from which you are upgrading.

To run a script, enter the following:

```
SQL> @uold_release.sql
```

Table 3-3 Upgrade Scripts

Old Release	Run Script
8.0.6	u0800060.sql
8.1.7	u0801070.sql
9.0.1	u0900010.sql
9.2	u0902000.sql

See Also:

["Determine the Upgrade Path to the New Oracle Database 10g Release"](#) if the old release you had installed prior to upgrading is not listed in [Table 3-3](#)

Make sure you follow these guidelines when you run the script:

- You must use the version of the script supplied with the new release 10.1 installation.
- You must run the script in the new release 10.1 environment.
- You only need to run one script, even if your upgrade spans more than one release. For example, if your old release was 8.1.7, then you only need to run `u0801070.sql`.

The upgrade script creates and alters certain data dictionary tables. It also upgrades or installs the following database components in the new release 10.1 database:

- Oracle Database Catalog Views
- Oracle Database Packages and Types
- JServer JAVA Virtual Machine

- Oracle Database Java Packages
- Oracle XDK
- Oracle Real Application Clusters
- Oracle Workspace Manager
- Oracle interMedia
- Oracle XML Database
- OLAP Analytic Workspace
- Oracle OLAP API
- OLAP Catalog
- Oracle Text
- Spatial
- Oracle Data Mining
- Oracle Label Security
- Messaging Gateway

14. Turn off the spooling of script results to the log file:

```
SQL> SPOOL OFF
```

Then, check the spool file and verify that the packages and procedures compiled successfully. You named the spool file in Step [12](#); the suggested name was `upgrade.log`. Correct any problems you find in this file and rerun the appropriate upgrade script if necessary. You can rerun any of the scripts described in this chapter as many times as necessary.

15. Run `utlu101s.sql`, specifying the `TEXT` option:

```
SQL> @utlu101s.sql TEXT
```

The Post-upgrade Status Tool displays the status of the database components in the upgraded database. The Upgrade Status Tool displays output similar to the following:

```
Oracle Database 10.1 Upgrade Status Tool MM-DD-YYYY HH:MM:SS
--> Oracle Database Catalog Views      Normal successful completion
--> Oracle Database Packages and Types Normal successful completion
--> JServer JAVA Virtual Machine       Normal successful completion
--> Oracle XDK                         Normal successful completion
--> Oracle Database Java Packages      Normal successful completion
--> Oracle Real Application Clusters   Normal successful completion
--> Oracle interMedia                 Normal successful completion
--> Oracle Text                       Normal successful completion
```

16. Shut down and restart the instance to reinitialize the system parameters for normal operation. The restart will also perform release 10.1 initialization for JServer JAVA Virtual Machine and other components.

```
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP
```

Cleanly shutting down and restarting the instance flushes all caches, clears buffers, and performs other housekeeping activities. These measures are an important final step to ensure the integrity and consistency of the newly upgraded Oracle Database.

Also, if you encountered a message listing obsolete initialization parameters when you started the database in Step [10](#), then remove the obsolete initialization parameters from the parameter file now.

17. Run `utlrp.sql` to recompile any remaining stored PL/SQL and Java code.

```
SQL> @utlrip.sql
```

Verify that all expected packages and classes are valid:

```
SQL> SELECT count(*) FROM dba_objects WHERE status='INVALID';
SQL> SELECT distinct object_name FROM dba_objects WHERE status='INVALID';
```

18. Exit SQL*Plus.

Your database is now upgraded to the new Oracle Database 10g release. Complete the procedures described in [Chapter 4, "After Upgrading a Database"](#).

WARNING:

If you retain the old Oracle software, then never start the upgraded database with the old software. Only start the database with the executables in the new release 10.1 installation directory. Also, before you remove the old Oracle Database environment, make sure you relocate any datafiles in that environment to the new Oracle Database environment. See the [Oracle Database Administrator's Guide](#) for information about relocating datafiles.

Troubleshooting the Upgrade

There are three resources that generally require increases for a new Oracle Database release:

- SYSTEM tablespace
- Shared memory
- Rollback segments

If you run out of one of these resources during the upgrade, then increase the resource allocation and re-run the appropriate upgrade scripts.

SYSTEM Tablespace

Typically you will receive one of the following messages during the upgrade if your SYSTEM tablespace size is insufficient:

```
ORA-01650: unable to extend rollback segment string by string in tablespace string
ORA-01651: unable to extend save undo segment by string for tablespace string
ORA-01652: unable to extend temp segment by string in tablespace string
ORA-01653: unable to extend table string.string by string in tablespace string
ORA-01654: unable to extend index string.string by string in tablespace string
ORA-01655: unable to extend cluster string.string by string in tablespace string
```

Shared Memory

You will require larger shared memory pool sizes, particularly if you have JServer in the database. The error message will indicate which shared memory initialization parameter needs to be increased.

```
ORA-04031: unable to allocate string bytes of shared memory ("string","string","string","string")
```

Refer to [Oracle Database Reference](#) for information about shared memory initialization parameters.

Public Rollback Segment

If you are using rollback segments, then you need to have a single large (70 MB) PUBLIC rollback segment online while the upgrade scripts are being run. Smaller public rollback segments should be taken offline during the upgrade. Typically you will get the following error if your rollback segment size is insufficient:

```
ORA-01562: failed to extend rollback segment number string
```

Abandoning the Upgrade

If you completed the steps in "[Backup the Database](#)" to back up your database, then the easiest way to abandon the upgrade is to restore that backup. Complete the following steps:

1. Log in to the system as the owner of the Oracle home directory of the previous release.
2. Sign on to RMAN:

```
rman "target / nocatalog"
```

3. Issue the following RMAN commands:

```
STARTUP NOMOUNT
RUN
{
  REPLICATE CONTROLFILE FROM 'save_controlfile_location';
  ALTER DATABASE MOUNT;
  RESTORE DATABASE FROM TAG before_upgrade
  ALTER DATABASE OPEN RESETLOGS;
}
```

[Previous](#)[Next](#)

ORACLE
Copyright © 2002, 2004, Oracle. All rights reserved.

[Home](#)[Book List](#)[Contents](#)[Index](#)[Master Index](#)[Feedback](#)