

# Installation, Server Configuration, and Database Upgrades

## CERTIFICATION OBJECTIVES

- |      |  |      |  |
|------|--|------|--|
| I.01 | Installation New Features Support                    | I.06 | Using New Utility to Perform Pre-Upgrade Validation Checks |
| I.02 | Performance Enhancements to the Installation Process | I.07 | Using the Simplified Upgrade Process                       |
| I.03 | Simplified Instance Configuration                    | I.08 | Starting Up the Database Using a New Mode When Upgrading   |
| I.04 | Viewing Database Feature Usage Statistics            | ✓    | Two-Minute Drill   |
| I.05 | Supported Upgrade Paths to Oracle Database 10g       | Q&A  | Self Test  |

**O**racle Corporation has ambitiously positioned Oracle Database 10g as a sophisticated self-managing database. Note that you don't refer to the new 10g database as Oracle10g, as in Oracle9i. The correct name of our new database is Oracle Database 10g. Throughout this text, I refer to the database as Oracle Database 10g, although I may occasionally refer to it as simply Oracle 10g, for convenience.

The database server I use in this book is the Oracle Database 10g Enterprise Edition Release 10.1.0.2.0. I use both a UNIX platform and a Windows platform for most of the examples in this book. You can probably replicate several examples in this book using a different version of the Oracle software, but you can get the Enterprise Edition software free from Oracle, so there isn't any reason for not using that version. Practicing using the new features is essential not only for passing the test, but also for learning the features well enough to implement them in your databases. So, if you haven't already downloaded the software, please do so now (from [www.oracle.com](http://www.oracle.com)).

The *g* in Oracle 10g, stands for *grid computing*. Grid computing is all the rage now, with IBM promoting “on-demand” computing, and Sun, HP, and other companies talking about their own visions of grid computing. Regardless of what a particular company might name it, grid computing really means treating computing as a utility, somewhat like an electrical company's power grid. When you want more or less computing power, the grid will supply it to you automatically in a flexible manner. The following three attributes lie at the heart of grid computing:

- *Virtualization* between the layers of the computing stack and the users
- *Dynamic provisioning* of work among the available resources, based on changing needs
- *Pooling* of resources to maximize availability and utilization

Several new manageability-enhancing features, along with revamped older features, promise Oracle database administrators (DBAs) a very new way of managing their powerful Oracle databases. The main goals of this new approach are to reduce administrative effort and capital expenditures, while simultaneously keeping failure costs to a minimum. The changes start, as they should, right with the installation of the database software itself.

This first chapter discusses in detail features related to installation, configuration, and upgrading.



***Please refer to the Oracle Database 10g documentation for complete details on a given feature. This book is intended to provide the information you need for upgrading your Oracle OCP certification from Oracle9i to Oracle Database 10g. (The complete name of the test required for upgrading your certification is Oracle Database 10g: New Features for Administrators (Exam #1Z0-040) Therefore, unless the feature is completely new to Oracle 10g, I don't attempt to fully explain it in this book.***

Let's start our review of the new Oracle 10g installation features by looking at some interesting changes in the installation process.

## CERTIFICATION OBJECTIVE 1.01

# Installation New Features Support

While the essential database software installation is through the familiar Oracle Universal Installer, there are several appealing changes in the Oracle Database 10g installation process itself. The entire installation process is simpler now, and the typical server installation takes less than a half an hour to complete. Client installations can now be done in less than a minute, and they require less than 70MB of disk space.

The following are some new features related to the Oracle Enterprise Server software installation process:

- Database management choices
- Automatic pre-install checks
- New file storage options
- Backup and recovery options
- Database user password specification
- Cluster Ready Services
- Enterprise Configuration Management Tool
- OracleMetaLink integration
- Oracle software cloning
- Database cloning

Let's look at each of these installation new features, starting with enhancements in database management choices.

### Database Management Choices

As part of the Oracle Database 10g software installation, Oracle offers you the choice to create a “starter” database, as was the case in previous versions. However, now, you have more management choices to make when you decide to let Oracle create the starter database.

In Oracle9i, you could create the Oracle Enterprise Manager (OEM) central management setup (including database management, as well as host and network management) as part of the normal database installation. In Oracle 10g, if you wish to configure a centralized (or grid-based) OEM capability, you must configure this feature separately.

In Oracle9i, you needed to choose between an OEM console in the stand-alone mode or log in to the Oracle Management Server. In the stand-alone mode, you could manage only your databases. In order to manage HTTP servers, events, and jobs, you needed to use the Oracle Management Server. You can now use the OEM in two fundamentally different ways:

- You can manage your databases *locally* using the *OEM Database Control*, which is part of the Oracle 10g server software.
- You can manage your databases *centrally*, through the *OEM Grid Control*, which is available on separate CDs.

#### exam Watch

#### ***The OEM Database Control comes with the Oracle Database 10g server software.***

The OEM Database Control is a web-based application that you can use to manage a single Oracle 10g database. The Database Control simply lets you monitor and manage a single database. You can access the Database Control by using a web browser; no configuration is necessary to start using the Database Control. Oracle automatically installs the OEM Database Control in the same Oracle Home as your Oracle database.

Alternatively, you can install the OEM Grid Control tool, which lets you monitor and manage several databases and servers at once. In order to use the OEM Grid Control, you need to configure and start the Oracle Management Agent on each of the nodes you wish to monitor.


**exam**  
**Watch**

**If you are managing a single database, the simple stand-alone OEM Database Control is adequate. On the other hand, if you are in charge of several databases on several hosts, you'll need to install and use the OEM Grid Control. The Grid Control includes the Oracle Management Agent, Oracle Management Service, the Oracle Management Repository, and the Grid Control console. It is through the console that you perform all monitoring, configuration, and management tasks for not only databases, but also for the various hosts and other services.**

In Oracle9i databases, there was only one kind of OEM, and you had the option of configuring it with or without Management Agents and the Management Repository. Now, Oracle is making a formal distinction between a local and centrally managed OEM.

The OEM Database Control runs with the help of a HTTP server, and you use a regular Internet browser to see the OEM interface. The default URL for the OEM Database Control is as follows:

```
http://host.domain:5500/em/
```


**on the**  
**Job**

In order to manage your database, log in as the user SYS and connect as SYSDBA.

**The default port number for the OEM is 5500. You can determine the correct port number for your database by looking at the portlist.ini file, which you can find in the \$ORACLE\_HOME/install directory.**

In order to run the OEM Database Control, you must first start the database from which the Database Control is running. Let's say your database name is *nina*. You then need to make sure that the service (on a Windows server) *OracleServiceDBConsoleNina* is running, before you try to access the OEM Database Control interface through your web browser.

When you choose local management using the Database Control, the installer also gives you the choice of configuring default backup strategies administered by the Recovery Manager (RMAN). In addition, choosing the Database Control also enables you to configure the sending of automatic e-mail alerts to warn you about space concerns and other problems.


**exam**  
**Watch**

**The Apache HTTP Server isn't a part of the Oracle Database 10g software, unlike in the previous versions. Oracle continues to provide the Apache HTTP Server, but as part of the supplemental product CD-ROM.**

The Database Configuration Assistant (DBCA) can automatically configure the OEM Database Control interface during database creation. However, if you create a database manually, you must configure and install the OEM Database Control, using the Oracle-supplied build script: `$ORACLE_HOME/bin/emca` for UNIX and LINUX systems and `$ORACLE_HOME\bin\emca.bat` for Windows. This script will configure the OEM repository and start up the Database Control console.

In order to access the OEM Database Control from your browser, you must first have the `dbconsole` process running on your system. The `dbconsole` process is automatically started during the installation process when you create a database using the DBCA. When you create a database manually, the `emca` (UNIX) or the `emca.bat` (Windows) script will start the `dbconsole` process. You can start, stop, and check the status of the `dbconsole` process by using the following three commands, respectively:

```
$ emctl start dbconsole
$ emctl stop dbconsole
$ emctl status dbconsole
```

### Automatic Pre-Install Checks

In order to install Oracle 10g, you must meet several operating system hardware and software requirements. In previous Oracle software versions, it wasn't uncommon for DBAs to start the installation process, only to stop it midway because they overlooked one or two factors. The DBA was responsible for making sure that there were enough system resources, that the correct set of initialization parameters were chosen, and so on. To make the requirements phase go smoothly, the Oracle Universal Installer (OUI) now manages the entire pre-install requirements check automatically. Oracle writes the results of the prerequisite checking process to a *results file*.

Comprehensive hardware and software prerequisite checks ensure the detection of resource and compatibility issues before your installation process gets underway. Oracle even allows for the updating of the prerequisites directly via *OracleMetaLink*, to reflect new operating system patches on your system. You may also add custom prerequisite checks through XML files, in order to customize the installation process.

### Pre-Installation Checks Performed

The following are some of the main pre-installation checks that the prerequisite checker of the OUI automatically performs before you install the Oracle 10g server software:

- **Correct operating system version and compatibility level** The first thing that Oracle checks before you start the installation process is whether you

are running the current operating system version. For example, on a HP-UX system, Oracle will verify that you have at least the HP 11.11 version. Similarly, for Linux-based systems, Oracle will make sure you have one of UnitedLinux 1.0, Red Hat 2.1, or Red Hat 3.0 versions before you can proceed further with the installation. In addition, if you have a 32-bit operating system, you cannot install the 64-bit version. Oracle will also ensure that you have set the database compatibility level to 9.2.0 or higher.

- **Operating system patches** Oracle automatically checks to make sure you have installed all the necessary operating system patches.
- **Kernel parameters** Oracle checks to make sure that operating system kernel parameters like SHMMAX and SHMMIN are set appropriately.
- **Sufficient memory and file space** Oracle checks the amount of swap space and the minimum space in the temporary tablespace that are required during the installation process. Oracle will also ensure that your redo logs are sized sufficiently.
- **Oracle Home** Your Oracle Home must be either empty or belong to a supported release that permits the installation of Oracle 10g server software.

## Methods for Performing Pre-Installation Checks

You can instruct the OUI to perform the prerequisite checks in three different ways:

- As part of an actual installation process
- As a stand-alone check without any installation
- From the command line, in the silent mode (without using the OUI GUI)

Only after you pass all the preinstall validations will the OUI proceed with the installation of the Oracle software.

## New File Storage Options

The OUI now offers three choices for configuring the file systems for any new starter database that you may create:

- **Automatic Storage Management (ASM)** This option offers a way to easily manage your datafiles using logical file management techniques. You can think of ASM as an integration of a traditional file system with a built-in Logical Volume Manager (LVM), which is separate from any LVM your

system may already have, like Veritas, for example. This means that you no longer need to use a third-party LVM to stripe and mirror your hard drives. Under ASM, all you need to do is create disk groups, and the database automatically stripes and mirrors your data across the available disks in the disk groups. You can also add or remove disks while the database is up and running.

- **Raw Devices** You need to think about using raw devices only if you are considering using Oracle Real Application Clusters. If you use RAC, and a Clustered File System (CFS) is available on your operating system, Oracle recommends using either CFS or ASM for your file storage. If a CFS is unavailable, Oracle recommends that you use raw, or “uncooked,” file systems or ASM.
- **File Systems** Choosing this option will mean that you are using the traditional operating system files and directories for your database storage. Of course, you should organize your directories and files according to the well-known Optimal Flexible Architecture (OFA) guidelines.

### Backup and Recovery Options

The next major management option involves the choice of backup and recovery options for the starter database, during the Oracle server software installation process. You have a choice of two options:

- Do not enable automatic backups
- Enable automatic backups

If you choose to enable automatic backups, you must also specify the location in which you want the backups to reside. In addition, you need to provide the *credentials*—name and password—for the user who will be performing the backups.



***You can enable the automated backups after you’ve finished the server software installation.***

### Database User Password Specification

The starter database will have several schemas for database management and post-installation functions: SYS, SYSTEM, DBSNMP, and SYSMAN schemas. Oracle



will leave the passwords for these four usernames unlocked and unexpired. During installation, you'll be asked to furnish passwords for these four accounts, and you can use the same password for all of them if you wish.

All other preloaded Oracle schemas will have passwords that are *expired and locked* when you create any new database. It's your job to unlock these standard user accounts and set new passwords for them.

In Oracle9i, you had to change the SYS and SYSTEM passwords twice: once during the installation of the software and once during database creation. In Oracle 10g, you need to provide this information only once, during the installation of the software.

## Cluster Ready Services

The Oracle 10g installation supports several Real Application Clusters (RAC) features, including the installation of the Cluster Ready Services (CRS) feature. CRS facilitate the management of RAC and replace third-party cluster management software on most operating system platforms. CRS provide a platform for services on RAC. For example, when an RAC component database goes down, CRS will automatically redirect the processing to an alternate available instance.



***You can invoke the OUI from any node on a cluster that is part of an installation. You can not only install new software, but also perform upgrades and apply patches from any node in a cluster***

## Enterprise Configuration Management Tool

The Enterprise Configuration Management Tool is available through the OEM Grid Control. As enterprises grow larger, so do the number and complexity of the hosts and databases you need to manage the enterprise. You can use the new OEM Grid Control tool to simplify the configuration and management of the enterprise. The Grid Control offers a management console for your entire system, and it enables you to monitor and configure complex enterprises by performing the following two tasks:

- Collect necessary host and database configuration information.
- Perform configuration changes.

Let's briefly look at how you can perform these tasks in your enterprise.

## Collecting Host and Database Configuration Information

The Oracle 10g OEM contains powerful new configuration management features to help enforce best practice configuration parameter settings and track configuration changes. These features apply to Oracle databases, as well as to the host servers and the network. Continuous configuration monitoring helps you to adhere to best practice configuration management and promotes availability, performance, and security.

The OEM Grid Control manages hosts and databases with the help of Management Agents that run on various hosts. These Management Agents collect information about the hosts, databases, listeners, web servers, and other services that run on the host. By default, the Management Agents collect and send information every 24 hours. The hosts send this information to the Oracle Management Repository. The following is a list of some of the host and database configuration information that the Management Agents send to OEM:

- Memory, CPU, and I/O configuration on the host
- Operating system details such as vendor, version, installed software components, patches, and database patch information
- Database and instance properties, including information about initialization parameters
- Tablespace and datafile information
- Control file and redo log information

The OEM maintains a detailed *configuration inventory* of all the databases and servers in its domain.

## Monitoring and Managing Enterprise Configuration

Once the Grid Control collects the configuration information, you can use the same tool to monitor and manage the configuration. Here's a brief list of the various things you can do to manage enterprise configuration with the OEM Grid Control:

- View and compare the hardware and software configurations of individual hosts and databases.
- Track changes to host configurations.
- See a summary view of your enterprise configuration.
- Search the enterprise configuration to find configurations that meet a set of search criteria that you specify.

- Clone Oracle Homes and database instances.
- View host and database policy violations.
- Simplify the process of patching Oracle products and manage Oracle critical patch advisories.

## MetaLink Integration

Probably every Oracle DBA knows and relies on the Oracle*MetaLink* service to some extent during the performance of his or her duties. In Oracle 10g, you can directly link the OEM to the Oracle*MetaLink* service. Through this built-in *MetaLink* integration, OEM can then automatically track any new software patches for you. You can arrange to receive alerts whenever the OEM spots new patches. The OEM can even tell you which systems need the new patches. Using the OEM Oracle*MetaLink* link, you can search for appropriate patches and download them, receive critical patch alerts, and stage and apply patches on your system.

How does Oracle keep track of the exact software version that you are using, if you are constantly downloading and uploading patches? Whenever you apply a patch, Oracle immediately updates the OUI's inventory, so it has the correct record of your latest patch level. Exercise 1-1 shows how to use the OEM Database Control to download Oracle software patches.

### EXERCISE 1-1

#### Using OEM Database Control to Download Oracle Software Patches

The OEM uses Oracle*MetaLink* credentials to search for and download Oracle*MetaLink* patches. If you did not specify your Oracle*MetaLink* credentials during installation, you can follow the steps in this exercise.

1. On the OEM Database Control home page, click Setup.
2. On the Setup page, click Patching Setup.
3. Specify your Oracle*MetaLink* username and password in the fields provided.
4. The URL to access the Oracle*MetaLink* web site to search for and download patches is displayed in the Patch Search URL field on this page, as shown here:

Patch Search URL : <http://updates.oracle.com>

You may access *OracleMetaLink* directly by going to the following web site:

<http://oracle.com/support/metalink/index.html>

5. From this page, Oracle licensees can register for an account or log in with an existing account. Once logged in, you can search for patches, download patches, and so on.

---

The OEM automatically alerts you to any new critical patches that become available. In addition, you can use the OEM Patch Wizard to find out what interim patches are available for your system. You can now study patch details and README notes directly from your OEM interface. Using the Patch Wizard, you can download available patches from *Metalink* into the OEM patch cache, so you can propagate these patches across your system at your convenience.

### Oracle Software Cloning

The OEM Grid Control enables you to easily duplicate Oracle Database 10g software installations (Oracle Homes) from a master installation to one more servers. Using an OEM Wizard, you can specify the source and the destination host to which you want to clone that Oracle Home. The cloning process will automatically adjust host names, IP addresses, listener settings, and so on.

You can create multiple new installations in a single operation from the OEM Grid Control framework. You can also create a library of master installations, which you can use repeatedly in cloning operations.

### Database Cloning

Using the OEM, you can now easily clone databases. OEM performs database cloning by using RMAN. You use the OEM Clone Database wizard, also known as the Clone Database Tool, to perform the various steps in a database cloning operation. To create a duplicate of an existing database, from the home page of the OEM Database Control, choose Maintenance | Deployments | Clone Database. OEM will back up the source database's datafiles as well as archive logs, and automatically restore and recover these in the target location. The OEM also creates the necessary password files, initialization files, and networking files.

When you use the OEM to clone a database, you can leave the database open. You can clone databases that are release 8.1.7 and higher. The new database will

be up-to-date with the old one until the time the archived logs of the original databases were backed up by the OEM.

## CERTIFICATION OBJECTIVE 1.02

# Performance Enhancements to the Installation Process

Oracle has implemented several changes to make the installation process more streamlined and faster than in previous versions. You can literally install the entire Oracle 10g software in less than half an hour, using just one CD. Moreover, during the installation process, you can now automatically set up several features, such as database monitoring. The important installation process enhancements are discussed in the following sections.

## Single CD Installation

Although the Oracle Database 10g server software comes in a pack of CD-ROMs, you need only a single 650MB CD to complete your Oracle 10g server installation. It takes only about 20 minutes to complete the entire installation. The label on this main database server software CD is *Oracle Database 10g CD*. Several older components are now available on separate CDs. Along with the Oracle Database 10g CD, the following CDs are supplied with the Oracle 10g package:

- Companion CD
- Oracle Database 10g Products CD
- Oracle Database 10g Client CD
- Oracle Documentation 10g CD
- Oracle Database 10g Demos CD
- Oracle Enterprise Manager CD

You install other tools, such as the HTTP Server, from the Oracle Database 10g Companion CD. The Companion CD includes two products: the HTTP Server and the new HTML DB feature.

The Database 10g Products CD includes the Oracle database examples, the Oracle Workflow Server, and the Legato Single Server Version (LSSV).

The original installation includes only the new OEM Database Control, which enables you to monitor just the database itself. The OEM Grid Control, which enables the monitoring of your entire system, is available on the Oracle Enterprise Manager CD.

### Minimal Hardware Requirements

You can use the Oracle Database 10g software with less stringent hardware requirements than those in previous versions. The following are the basic hardware requirements for Oracle Database 10g:

- **Memory** You need 256MB for the basic database, and 512MB if you are using the stand-alone version of the OEM (the OEM Database Control). If you need to, you can run both the database and Database Control with 256MB or even less memory, say, for testing purposes. However, the more memory you have, the easier it is to use the memory-hungry, Java-based OEM tool.
- **Disk space** You need a maximum of about 2.5GB of disk space for the Oracle software. In addition, you need 1GB of swap space and about 400MB of disk space in the /tmp directory. The OUI uses a more sophisticated algorithm to estimate your disk space needs now, so there is less chance of running out of disk space during the installation process.

### Easier and Cleaner Deinstallation

Prior to Oracle Database 10g, you couldn't always count on a clean Oracle server deinstallation if you needed to remove the server software for some reason. Well, this isn't true anymore. With Oracle 10g, you can perform an easy and clean deinstallation, marked by the following new features:

- All files, including files that belong to configuration assistants and patch sets, are automatically removed during the deinstallation process.
- The installer automatically cleans up all the Windows Registry entries.

To deinstall your Oracle 10g software, follow these steps:

1. First, shut down all databases and ASM instances running under the Oracle Home you want to remove, and then remove the databases (remove all directories with database files).
2. Stop all the relevant processes running under this Oracle Home, by running the following commands:

```
$ORACLE_HOME/bin/emctl stop dbconsole - shuts down the OEM.
$ORACLE_HOME/bin/lsnrctl stop - brings down the Oracle listener
$ORACLE_HOME/bin/isqlplusctl stop - brings down the iSQL*Plus server.
```

3. Start the OUI.
4. Click Deinstall Products in the Welcome window.
5. In the Inventory window, select the correct Oracle Home that contains the software you want to deinstall, and then click Remove.
6. Manually remove the Home directory that you just deinstalled.



**Don't remove any Oracle Home directories without first using the OUI to remove the Oracle software.**

## Automatic Launching of Software

In Oracle 10g, several Oracle products will launch automatically immediately after you complete the server installation. Examples include the Oracle Management Agent, the OEM Database Control, and the iSQL\*Plus server. Oracle automatically sets up an out-of-the-box monitoring and administration environment for you after the installation is complete.

## Response File Improvements

The *response file* is a file you can use to perform *silent installations*, which are installations where the user doesn't need to provide any interactive input. The response file contains all the information that you normally provide during an interactive installation session. If you have several off-site Oracle installations to perform, you could send an installation response file to these locations, rather than going to each place to perform a manual installation.

The following are the new Oracle 10g improvements in the response file, which help you perform a truly “silent” Oracle installation:

- The file has a new header format, which makes the response file easier to edit.
- You don’t need to specify an X server when performing installations in a character mode console.
- You don’t need to set the `DISPLAY` variable on UNIX systems.
- No GUI classes are instantiated, making this a truly silent method of installing software.

## CERTIFICATION OBJECTIVE 1.03

### Simplified Instance Configuration

In Oracle Database 10g, several enhancements make instance creation a lot simpler than before. These enhancements include a more sophisticated DBCA, adoption of a set of simplified database initialization parameters, and the ability to view database feature usage statistics. The following are the main new features related to simplified instance configuration:

- DBCA enhancements
- Policy-based database configuration framework
- Simplified initialization parameters
- Irreversible datafile compatibility

Let’s start with a discussion of the enhancements in the DBCA tool.

### Database Configuration Assistant (DBCA) Enhancements

You can create new databases manually by using the standard `create database` statement. You can also create databases with the help of Oracle’s DBCA. The DBCA really makes the Oracle DBA’s database-creation tasks tremendously lighter when creating complex Oracle 10g databases. Oracle 10g contains several enhancements in routine database administration, and the DBCA helps you implement all these new automatic management tasks. Using the DBCA ensures that you’re reminded



about all the important options, rather than needing to remember them and perform them all manually.

## EXERCISE 1-2

### Creating a New Database Manually

You can create a new database with the simple statement `create database abcd`, where `abcd` stands for your database instance name.

1. Create the instance before you try creating the database. On a Windows system, first run the command `oradim -new -sid my_sid`. Make sure you create a simple `init.ora` file with just one parameter: the instance name.
2. Log in as SYSDBA.
3. Use the command `create database my_sid`. Do not specify any datafiles, tablespaces, or any other information.
4. Answer the following questions:
  - What tablespaces does Oracle create? Where are the datafiles?
  - What are key features of the initialization parameter file (`init.ora` or in the `SPFILE`)?
  - How much SGA and other PGA memory is allocated?
  - What mode of undo management does your new database use?
  - What are the default passwords for the various system accounts?
  - What database features are currently configured?



***Please make sure you create the instance before you try creating the database. On a Windows system, first run the command `oradim -new -sid my_sid`. Make sure you create a simple `init.ora` file with just one parameter—the instance name. Next, log in as `sysdba` and use the command `'create database my_sid`. Now, you can answer the exercise questions.***

There are several important DBCA enhancements in Oracle Database 10g, including the creation of the mandatory SYSAUX tablespace, a flash recovery area, and new database storage options. Let's start with a discussion of the new mandatory SYSAUX tablespace.

## The SYSAUX Tablespace

You are aware that the OEM, the RMAN, and several other Oracle tools need repositories—locations where these tools store all the database metadata. The SYSTEM tablespace stores a lot of metadata, such as the data dictionary, but it does not store the metadata relating to tools like the OEM. Traditionally, Oracle required you to create separate tablespaces for the various repositories in your database. Now, in Oracle 10g, Oracle introduces the SYSAUX tablespace, which is auxiliary to the SYSTEM tablespace. This new tablespace is a central location for the metadata of all tools like the OEM and RMAN. The SYSAUX tablespace consolidates the data that various database features and products used to store in multiple tablespaces (or in the SYSTEM tablespace). Thus, you need to create fewer tablespaces to store all the repository data for the various utilities like the RMAN.

The DBCA automatically creates the SYSAUX tablespace for you, with a default size of 500MB. Of course, you can customize the storage and file parameters for SYSAUX, if you wish. Toward the end of this chapter, in the section detailing the manual upgrading of a database to the Oracle 10g version, I'll show you the syntax for manually creating the SYSAUX tablespace.

## Flash Recovery Area

The *flash recovery area* is a unified storage location on your server that Oracle reserves exclusively for all database recovery-related files and activities. The main purpose behind the creation of the flash recovery area is to automate the storage management for many backup and recovery-related activities. The database will store all the files that it needs during a database recovery in this flash recovery area.

Note that the flash recovery area is separate from the space you allocate to the normal database files—datafile, redo log files, and the control files. You must allocate separate physical space for the flash recovery area. When you use the DBCA to create a new database, it automatically creates the flash recovery area for you.



***The flash recovery area is mandatory if you wish to configure automatic backups using the OEM.***

How do you determine what size your database flash recovery area should be? Oracle recommends that you allocate at least enough space to hold a copy of the following files:

- All datafiles
- All redo log files

- All control files
- All archived redo logs necessary for recovery that haven't been migrated to tape
- Any incremental backups, if they exist

If you want to manually define a flash recovery area, you can do so using the new flash recovery initialization parameters, `DBA_FLASH_RECOVERY_DEST` and `DB_FLASH_RECOVERY_DEST_SIZE`, as follows:

```
SQL> ALTER SYSTEM SET db_recovery_file_dest_size = 500G;
SQL> ALTER SYSTEM SET db_recovery_file_dest = '/u22/oradata/
finance/recovery';
```

You can dynamically alter or disable both of the flash recovery initialization parameters. You can find out details about the flash recovery area's contents by using the `V$FLASH_RECOVERY_FILE_DEST` view, as shown here:

```
SQL> SELECT name, space_limit, space_used,
           space_reclaimable, number_of_files
           FROM v$recovery_file_dest;
```

## Automatic Storage Management (ASM)

ASM is an exciting new Oracle Database 10g feature. In essence, an ASM-based storage system contains its own LVM for Oracle files. ASM makes it easier to manage large databases. Chapter 10 is devoted to a detailed discussion of the ASM feature.

## Management Options

As in the case of the starter database described earlier in this chapter, the DBCA gives you a choice between two kinds of database management: local and central. If you want to just manage a single database instance, simply choose the Database Control, which Oracle automatically installs with your new database. If, on the other hand, you need to manage several databases and several nodes, use the new Grid Control. The Grid Control doesn't come automatically with the database; you need to install this tool from a separate CD.

As mentioned earlier in this chapter, Oracle automatically sets up an out-of-the-box monitoring and administration environment for you after the installation is complete. You don't need to perform any configuration to start using the DB Console (Database Control) to start managing a single database, including alert notification, job scheduling, and software management.

You can access the OEM Database Control or the Grid Control from any client browser. For example, to access the OEM Database Control, just open Internet Explorer or Netscape and type in the following URL:

```
http://myhostname:default_port_number/em
```

where *myhostname* stands for the IP address or symbolic name of your machine, and *default\_port\_number* is the port number that Oracle allocated to the Database Control during installation. The default Database Control HTTP port number is 5500.

Once you get to the Database Control login page, log in as the SYS user (with the SYSDBA password). This will take you to the home page of the OEM Database Control utility.

Note that you'll need Oracle licensing for the following three premium components of OEM Database Control (technically speaking, you'll be in violation of your Oracle licensing requirements, if you use these options without paying for them!):

- Database Diagnostics Pack
- Database Tuning Pack
- Configuration Management Pack

Here's a brief listing of the components of each of these OEM premium packs:

Database Diagnostics Pack	Database Tuning Pack	Configuration Management Pack
Performance Monitoring (database and host)	SQL Access Advisor	Database and Host Configuration
Automated Database Diagnostic Monitor (ADDM)	SQL Tuning Advisor	Deployments
Automatic Workload Repository	SQL Tuning Sets	Patch Database and Patch Staging
Event Notifications: Notification Methods, Rules, and Schedules; Notification Blackouts	Object Reorganization	Clone Database and Clone Oracle Home
Event history/metric history (database and host)		Search and Compare Configuration Policies

## Policy-Based Database Configuration Framework

Oracle provides a set of established *configuration recommendations*, which are a set of Oracle's best practice recommendations, to ensure that your database performs at an optimal level. Oracle 10g enables you to monitor all of your databases to see if there

are any violations of the predetermined configuration policies. Oracle collects these configuration metrics for databases, host machines, and listener services.

The OEM Database Control provides an easy way to monitor the adherence of your system to the predetermined policies. On the Database Control home page, there is a section called Diagnostic Summary, which shows you if there are any policy violations anywhere. If you drill down, using the All Policy Violations button, you can get to the Policy Violations page, which summarizes all policy violations in your databases and hosts. If you wish, you can disable a policy by going to the Manage Policy Library page.

Here are a few examples of some typical policy rules:

- The policy rule “critical patch advisories for Oracle Homes” checks for missing Oracle patches.
- The policy rule “insufficient number of control files” checks for the use of a single control file.
- The “listener password policy rule” checks for password-protected listeners.

## Simplified Initialization Parameters

Instance configuration includes the all-important choice of initialization parameters. Oracle 10g informally divides the available set of initialization parameters into a basic and an advanced set of parameters, as follows:

- **Basic initialization parameters** This set consists of about 25 to 30 of the most common parameters that you need for an Oracle database. Oracle defines the basic initialization parameters as those that are adequate to keep your database running well. Examples include parameters like `PROCESSES`, `SESSIONS`, `UNDO_MANAGEMENT`, `DB_BLOCK_SIZE`, `CONTROL_FILES`, and `DB_NAME`.
- **Advanced initialization parameters** These are parameters you’ll need to deploy only rarely, to improve your database’s performance or to overcome some special performance problems. Some examples of the initialization parameters in the advanced set are the `CURSOR_SHARING` and `CURSOR_SPACE_FOR_TIME` parameters. Oracle recommends that only experienced Oracle DBAs configure the advanced initialization parameters.

## Initialization Parameter Views

You can view the database initialization parameters, as in Oracle9i, by querying the `V$PARAMETER` view or the `V$SPPARAMETER` view, depending on whether you are using the `init.ora` file or the `SPFILE` to store your initialization parameters.

**EXERCISE 1-3****Using the OEM Database Control to View Initialization Parameters**

You can also use the OEM Database Control to view your initialization parameters. To view the parameters using the Database Control, follow these steps:

1. Go to the Database Control home page.
2. Click the All Initialization Parameters tab on the Administration page.
3. If you wish to change any parameters, you can do so from here and click Save to File.
4. If you wish configuration changes to come into force immediately, click the SPFILE tab first, and then check the Apply Changes in SPFile mode to the current running instance box.

**Changes in the Initialization Parameters**

In Oracle Database 10g, several traditional initialization parameters have become obsolete or deprecated. You also have access to several new parameters. Here's a list of some of the important deprecated, obsolete, and new initialization parameters in Oracle Database 10g:

Deprecated Parameters	Obsolete Parameters	New Parameters
MTS_DISPATCHERS	DISTRIBUTED_TRANSACTIONS	RESUMABLE_TIMEOUT
UNDO_SUPPRESS_ERRORS	JOB_QUEUE_INTERVAL	SGA_TARGET
PARALLEL_AUTOMATIC_TUNING	ORACLE_TRACE_COLLECTION_NAME	PLSQL_OPTIMIZE_LEVEL
	MAX_ENABLED_ROLES	

**Irreversible Datafile Compatibility**

In prior Oracle database versions, you could revert to a previous compatibility setting by using the `alter database reset compatibility` command. This command is now obsolete. Once you start the database, you cannot go back to a compatibility value that's less than the current value of the `COMPATIBLE`

initialization parameter in your init.ora (or SPFILE) file. Thus, the COMPATIBLE initialization parameter is now irreversible. This section provides the new details on the compatibility feature.

The *minimum value* of the compatibility setting is now 9.2.0. That is, if you set the COMPATIBLE initialization parameter to 9.2.0., your datafile formats will be compatible with the Oracle9i Release 2 version. This means that if you wish, you can always downgrade your new Oracle 10g database to a 9.2.0 version, as long as you have installed the Oracle9i Release 2 (9.2.0.3) or later executables on your system along with the Oracle 10g server software. If you set the COMPATIBLE initialization parameter to 9.2, however, you can take advantage of only a subset of the new Oracle 10g features. Therefore, use the 9.2 compatibility setting only if you think you may need to revert to the Oracle9i version for some reason.

By *default*, Oracle sets the file format compatibility to 10.0.0. If you don't set the COMPATIBLE parameter in your init.ora file, or if you set it explicitly to 10.0.0, the compatibility level will be set to the Oracle 10g file format compatibility. This means that you won't be able to downgrade the Oracle 10g database to a prior release. In that case, in order to start your database with a lower compatibility setting, you would need to perform a point-in-time database recovery and recover to a point in time that was before you advanced the compatibility level.

To check the current compatibility level of your database, which is set by the value you chose for the COMPATIBLE initialization parameter, use the following statement.

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

**exam**  
**Watch**

**The minimum value of the COMPATIBLE initialization parameter is 9.2.0. The default value, however, is 10.0.0.**

## CERTIFICATION OBJECTIVE 1.04

# Viewing Database Feature Usage Statistics

In Oracle 10g, you can track database usage metrics, which enable you to understand two important aspects:

- How you are using the various features of your Oracle database, including whether the database is currently using a given feature, as well as the first and last times it used a given feature.

- The high-water mark (HWM) statistics for important database attributes. The HWM is simply the highest usage point a feature has attained up to now.

The database features that you can track include features like Advanced Replication, Oracle Streams, Virtual Private Database (VPD), and various auditing features. Oracle Database 10g collects HWM statistics for items like the following:

- Maximum size of tables
- Maximum number of datafiles
- Maximum number of sessions
- Size of the largest data and index segment

Let's briefly look at how the Oracle database collects these statistics.

## The Statistics Collection Process

How exactly does Oracle collect these new kinds of statistics? Oracle Database 10g introduces a new database process called *Manageability Monitor Process* (MMON), which records both the database usage statistics and the HWM statistics for various objects. MMON is a new Oracle 10g background process, and it is primarily responsible for various manageability-related background tasks, including taking snapshots of data, issuing database alerts, and collecting statistics.

MMON records the various statistics inside the Automatic Workload Repository (AWR), which, as you'll see in later chapters, is a new Oracle Database 10g innovation that stores database performance data. By default, MMON samples the data dictionary once a week, to collect both database feature usage and HWM statistics.

You can query the new `DBA_FEATURE_USAGE_STATISTICS` data dictionary view to find out the usage statistics of various features that MMON has stored in the AWR. Here's the structure of this view:

```
SQL> desc dba_feature_usage_statistics
Name                               Null?      Type
-----
DBID                                NOT NULL   NUMBER
NAME                                NOT NULL   VARCHAR2 (64)
VERSION                             NOT NULL   VARCHAR2 (17)
DETECTED_USAGES                     NOT NULL   NUMBER
TOTAL_SAMPLES                       NOT NULL   NUMBER
CURRENTLY_USED                      VARCHAR2 (5)
FIRST_USAGE_DATE                    DATE
```



LAST_USAGE_DATE	DATE
AUX_COUNT	NUMBER
FEATURE_INFO	CLOB
LAST_SAMPLE_DATE	DATE
LAST_SAMPLE_PERIOD	NUMBER
SAMPLE_INTERVAL	NUMBER
DESCRIPTION	VARCHAR2 (128)

You can also use the procedures REGISTER\_DB\_FEATURE and REGISTER\_HIGH\_WATER\_MARK in the new DBMS\_FEATURE\_USAGE package to track feature usage and HWM statistics.

You can query the DBA\_HIGH\_WATER\_MARK\_STATISTICS data dictionary view to see the HWM statistics and a description of all the database attributes that the database is currently monitoring. Here's a simple example:

```
SQL> select name, highwater, description
      2* from dba_high_water_mark_statistics;
USER_TABLES      Number of User Tables                                760
SEGMENT_SIZE     Size of Largest Segment (Bytes)                    159383552
PART_TABLES      Max Number of Partitions belonging to a Table       0
PART_INDEXES     Max Number of Partitions belonging to a Index       0
USER_INDEXES     Number of User Indexes                             1353
SESSIONS         Max No of Concurrent Sessions seen in database      28
DB_SIZE          Maximum Size of the Database (Bytes)                901775360
DATAFILES        Maximum Number of Datafiles                          5
TABLESPACES      Maximum Number of Tablespaces                       6
CPU_COUNT        Maximum Number of CPUs                              1
QUERY_LENGTH     Maximum Query Length                                2122
Maximum Number of Services                            4
12 rows selected.
SQL>
```

In this section, I have shown how you can query various data dictionary views to track the database feature usage and HWM statistics. However, there's an easier way to monitor these statistics: just use the OEM Database Control to quickly check the statistics. The next section describes how to use the OEM Database Control to monitor database usage statistics.

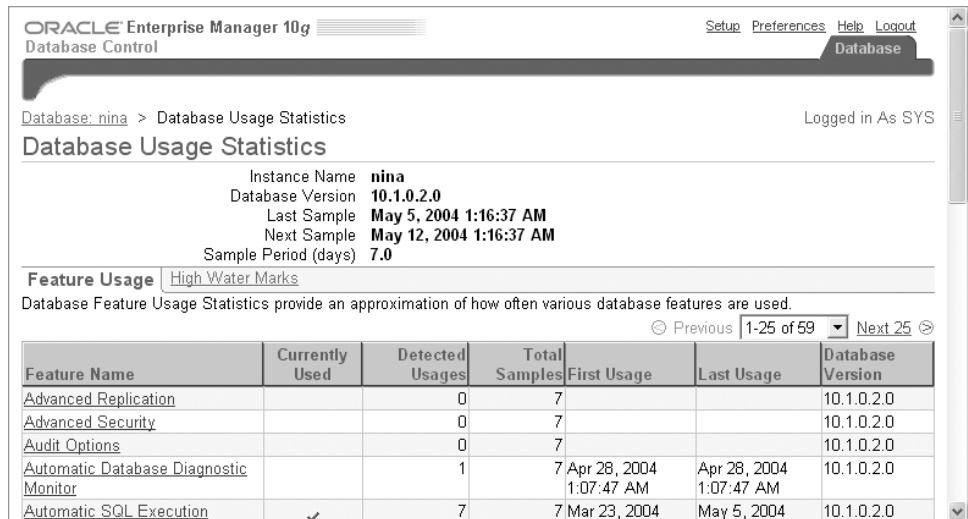
## Database Usage Statistics in the OEM

There are times when you would like to see whether and how the database is making use of important features. A *feature* is simply a specific area of functionality for the database. You can use the OEM Database Control to examine a list of the database

features and how the database is using those features. These usage patterns include the frequency of usage, the first time the database used a certain feature, and the most recent time the database used a certain feature. Following are the steps to viewing database usage statistics in the OEM Database Control:

1. Go the Database Control home page. Click the Administration link and go to the Configuration Management group. Click the Database Usage Statistics link.
2. You'll now be in the Database Usage Statistics property sheet. This property sheet leads to two pages: the High Water Mark page and the Feature Usage page. The Feature Usage page lists all the available database features by name. You can also see if the database is currently using a certain feature, as well as the first usage and last usage timings. To view details about the usage statistics of any feature, just click the associated link on this page. Figure 1-1 shows the Feature Usage page of the Database Usage Statistics property sheet.
3. If you want to view the database HWMs, click the High Water Marks tab in the Database Usage Statistics property sheet. This will take you to the High Water Marks page. Here, you can see the HWM attained for each database object, as well as the last sampled value for each feature and the version of the database feature. Figure 1-2 shows the High Water Marks page of the Database Usage Statistics property sheet.

**FIGURE 1-1**  
The Feature Usage page of the Database Usage Statistics property sheet



**FIGURE 1-2**

The High Water Marks page of the Database Usage Statistics property sheet

The screenshot shows the Oracle Enterprise Manager 10g interface. At the top, it says 'ORACLE Enterprise Manager 10g Database Control'. There are navigation links for 'Setup', 'Preferences', 'Help', and 'Logout'. Below that, it indicates 'Database Usage Statistics' for a database named 'nina'. The instance name is 'nina', the database version is '10.1.0.2.0', the last sample was taken on 'May 5, 2004 1:16:37 AM', and the next sample is scheduled for 'May 12, 2004 1:16:37 AM'. The sample period is 7.0 days. The user is logged in as 'SYS'. The 'High Water Marks' tab is selected, showing a table with columns for Name, High Water Marks, Last Sampled Value, and Database Version. The table lists various database metrics such as the number of CPUs, concurrent sessions, datafiles, partitions, services, tablespaces, query length, and total database size.

Name	High Water Marks	Last Sampled Value	Database Version
Maximum Number of CPUs	1	1	10.1.0.2.0
Maximum Number of Concurrent Sessions seen in the database	28	18	10.1.0.2.0
Maximum Number of Datafiles	5	5	10.1.0.2.0
Maximum Number of Partitions belonging to an User Index	0	0	10.1.0.2.0
Maximum Number of Partitions belonging to an User Table	0	0	10.1.0.2.0
Maximum Number of Services	4	4	10.1.0.2.0
Maximum Number of Tablespaces	6	6	10.1.0.2.0
Maximum Query Length	2122	1258	10.1.0.2.0
Maximum Size of the Database (Bytes)	1043333120	1043333120	10.1.0.2.0

## CERTIFICATION OBJECTIVE 1.05

# Supported Upgrade Paths to Oracle Database 10g

Several of you currently use Oracle8*i* or Oracle9*i* databases. You'll most likely be upgrading all these databases to the new Oracle Database 10g version in the near future. I have good news for all DBAs who are embarking on an Oracle Database 10g upgrade project: Oracle has made the upgrade process remarkably simpler by automating most of the upgrade process.

The following are the upgrade paths available to you to move up to Oracle Database 10g:

- If you have Oracle 7.3.3 or lower, or Oracle 7.3.4, 8.0.3, 8.0.4, 8.0.5, 8.1.5, or 8.1.6, you can't directly upgrade to Oracle Database 10g. You'll need to first upgrade to an intermediate release database.
- If you are using an Oracle 8.0.6, 8.1.7, 9.0.1, or 9.2 database, you can directly upgrade to Oracle Database 10g.

**exam**  
**Watch**

**You can migrate directly to the Oracle Database 10g version only if your database is one of the following versions: 8.0.6, 8.1.7, 9.0.1, or 9.2.**

You can upgrade to Oracle Database 10g in two ways: the traditional manual mode or by using the Database Upgrade Assistant (DBUA), a tool that automates the upgrade process. The DBUA is a refined version of the old Oracle Data Migration Assistant used in previous versions. Note that you can also use the traditional export and import utilities to perform your database upgrade, especially if you have a very small database.

Unlike in prior versions, the Oracle 10g upgrade process, even when you use the manual method, is somewhat automatic. As you'll see in the following sections, the manual process invokes the `startup upgrade` command, after which you need to run the main upgrade script, which performs all the necessary upgrades in the correct dependency order. Oracle will determine the upgrade order of the various components, by querying the new `DBA_SERVER_REGISTRY` data dictionary view. Oracle uses this view to check for the existence of the various components that it needs to upgrade. Oracle also queries the view to check the upgrade status of each component after the main upgrade script finishes running. The `DBMS_SERVER_REGISTRY` is also the basis for the new Post-Upgrade Status Tool, which you'll learn about in a later section.



***You can use either the `DBA_REGISTRY` or the `DBA_SERVER_REGISTRY` view to ascertain the upgrade status of individual database components loaded into the database. The two dictionary views are identical, except that that the `DBA_REGISTRY` has an extra column: `namespace`. If you query this view with the `namespace` set to `SERVER`, the results are identical to the results you'll obtain by using the `DBA_REGISTRY` data dictionary view.***

Traditionally, Oracle DBAs have needed to run a number of scripts during the database upgrade process, which made the whole process very strenuous and error-prone. In Oracle Database 10g, you can now run the entire upgrade process with a single upgrade script provided by Oracle. For example, to migrate from an Oracle 8.1.7.4 version database to the Oracle Database 10g version, you'll need to run the `u0801070.sql` script. You'll see a detailed example of the execution of this script in the “Steps in the Manual Upgrade Process” section later in this chapter.

In the following sections, I'll briefly explain the upgrade process using both the available methods.

## The Manual Upgrade Process

The manual upgrade process means you do all the due diligence work: make sure you remove or change all obsolete initialization parameters and run all the Oracle-provided database upgrade scripts. During a manual upgrade process, you, the DBA, must run SQL scripts from the command line to drive the upgrade process.

The advantage in using this method is that you control every part of the entire upgrade process. Of course, the drawbacks of a manual method are that you must perform a backup of the database yourself before the upgrade, remove or add all necessary initialization parameters, and ensure that the SYSTEM tablespace has adequate free space.

## The Database Upgrade Assistant (DBUA)

When you use the DBUA, the tool performs all the preinstallation checks for you to ensure that your database meets all the upgrade requirements, and then manages the upgrade process automatically. Here is a summary of what the DBUA does for you during a database upgrade:

- Perform all pre-upgrade tasks, including checking for invalid datatypes, desupported character sets, invalid user accounts, and sufficient free space in the tablespaces.
- Back up the database.
- Create any necessary objects.
- Invoke the correct upgrade script.
- Show the upgrade progress during the upgrade.
- Create new parameter and listener files in the new Oracle Home.



***The DBUA is a GUI tool, but you can also run it in the silent mode, by using the following command at the operating system level: `dbua`.***

To begin with, you need to analyze your existing system to see what changes may be necessary. Fortunately, Oracle provides an excellent script called the Upgrade Information Tool, which will ferret out this information for you automatically. We'll look at this new tool in the following section.

### CERTIFICATION OBJECTIVE 1.06

## Using New Utility to Perform Pre-Upgrade Validation Checks

Oracle now includes a brand-new tool, called the Upgrade Information Tool, to help you collect various pieces of critical information before you start the upgrade process.

Too often, as you are aware, you might need to restart an upgrade process, owing to incompatible initialization features or inadequate tablespace sizes.

## The Upgrade Information Tool

The Upgrade Information Tool provides the following important information:

- **The SYSAUX tablespace** You need to create a brand new tablespace called the SYSAUX tablespace before you can run the Oracle Database 10g upgrade script. The Upgrade Information Tool will tell you about this and recommend the correct size for this tablespace.
- **Information about log files** Oracle Database 10g requires the redo log files to be at least 4MB in size. If your current log files in the database you are going to upgrade are smaller than 4MB, the script will tell you to increase the size of the redo log files before the upgrade.
- **Tablespace sizes** If the current tablespaces don't have enough free space, the information will be logged, so you can increase the size of the tablespaces.
- **Initialization parameters** The pre-upgrade script will save you a lot of headaches by telling you which of your initialization parameters should be removed (deprecated and obsolete parameters) and which new parameters should be added before you can upgrade.
- **Database versions and compatibility level** The Upgrade Information Tool lets you know if you need to change your database compatibility level, using the COMPATIBLE initialization parameter.
- **Time estimates** The Upgrade Information Tool also provides you an approximate estimate of the time it will take for you to complete your upgrade.

As you can see, running the Upgrade Information Tool prior to your manual upgrade process does a lot of the manual work for you! As long as you read the log file of the Upgrade Information Tool carefully and implement the recommended changes, you are well set to upgrade to Oracle Database 10g.

**exam**  
**Watch**

***Both the manual upgrade process and the DBUA can use the Upgrade Information Tool. If you are performing a manual upgrade, you need to invoke the tool by running the SQL script utlu10i.sql. The DBCA automatically runs it as part of the pre-upgrade check.***

The Upgrade Information Tool is really nothing but a simple Oracle-provided SQL script, called `utlu101i.sql` (located in the usual `$ORACLE_HOME/rdbms/admin` directory), which helps you find out how your database stacks up for the upgrade process. Here's how you start the Upgrade Information Tool in your pre-upgrade database:

```
SQL> @$ORACLE_HOME\rdbms\admin\utlu101i.sql
Oracle Database 10.1 Upgrade Information Tool    04-14-2004 10:07:08
Database:
-----
-> name: FINANCE
-> version: 8.1.7.0.0
-> compatibility: 8.1.0
WARNING: Database compatibility must be set to 9.2.0 prior to upgrade.
Oracle Database 10.1 Upgrade Information Tool    04-14-2004 10:07:08
. ...
```

## The Post-Upgrade Status Tool

Oracle Database 10g also provides a new Post-Upgrade Status Tool, which gives you an accurate summary of the upgrade process and any necessary corrective steps to be taken. In the past, you had to determine the success of an upgrade by looking for any error messages. However, the absence of error messages during the upgrade doesn't guarantee that your upgrade was successful. The Post-Upgrade Status Tool looks in the component registry called `DBA_SERVER_REGISTRY` to check the status of each database component. If one or more components don't have a valid status or carry the wrong version number, the Post-Upgrade Status Tool will list the component information for you.



***You can restart a failed database upgrade job from the point where you failed.***

The Post-Upgrade Status Tool provides the following information:

- The name and status, either `VALID` or `INVALID`, of each database component
- The component's version compatibility with the current database version
- Suggestions for corrective action to take if there are any invalid components after the upgrade process is completed (such as the appropriate script names to run)

As in the case of the Upgrade Information Tool, you use an Oracle-supplied script to invoke the Post-Upgrade Status Tool. This script is the `utlu101s.sql` script, located in the `$ORACLE_HOME/rdbms/admin` directory on your server.

on the  
job

***if you use the DBUA to upgrade, the `utlu101s.sql` runs automatically. If you are performing a manual upgrade, you need to run the script yourself, after the upgrade process is finished.***

## CERTIFICATION OBJECTIVE 1.07

### Using the Simplified Upgrade Process

Although it is easy to perform a manual upgrade to Oracle Database 10g, the fact remains that you need to do all the due diligence, and a critical mistake like omitting a key step could cost you dearly in terms of time. Oracle recommends that you use the DBUA to facilitate the database upgrade process. The DBUA takes care of the following tasks for you:

- Deletes all obsolete initialization parameters
- Changes the `ORACLE_HOME` settings automatically
- Runs the appropriate upgrade scripts for your current release
- Configures your `listener.ora` file

Both the manual upgrade and the DBUA upgrade process perform the same set of steps: performing pre-upgrade checks to ensure that the database meets all the requirements, running the Oracle-supplied upgrade script, and so on. However, the DBUA performs additional chores like configuring the `listener.ora` file.

on the  
job

***Which one of the two methods is superior? I think either one is good enough, with the DBUA being easier on your nerves if you are relatively new to the database upgrading business. If you are a veteran Oracle DBA, going through the manual process may be the best way to go, in my opinion.***

Note that some of the material regarding the database upgrade process in the following sections may be more detailed than what's necessary to merely pass the certification exam. However, the review would come in handy when you are upgrading your databases to Oracle Database 10g.



## How Does the DBUA Work?

The DBUA performs several steps before and during the database upgrade process. In effect, the DBUA combines the work of the Upgrade Information Tool and the manual upgrade process, which you have seen in the previous sections. The DBUA performs all the necessary pre-upgrade steps for you automatically, including the following:

- Change the `ORACLE_HOME` settings to the new Oracle Database 10g software locations.
- Change the `/etc/oratab` entries to reflect the new location.
- Copy the existing `init.ora` file from its old location to the new Oracle Database 10g default `init.ora` location (`$ORACLE_HOME/dbs` under UNIX).
- Check for adequate space in your tablespaces, including the undo tablespace or the rollback segments, before the upgrade process begins.
- Check for desupported character sets, invalid data types, invalid usernames, and so on.
- Perform a backup of the database, if you choose.

As you have seen, the upgrade process isn't really a horrendous process, even when you do it using the manual method. With the DBUA, upgrading becomes even easier. As you know by now, the DBUA incorporates all three steps of the upgrade process: the pre-install checks, the actual upgrade process, and the post-upgrade checks. You can use the DBUA to upgrade any database configuration, including RAC and standby databases.

## DBUA Startup

You can start the DBUA by selecting Programs | Oracle | Configuration and Migration Tools | Database Upgrade Assistant. Figure 1-3 shows the Welcome screen of the DBUA. If you are on a UNIX system, simply type `dbua` (after logging in as the Oracle user) at the operating system prompt to start up the DBUA GUI.

If you don't want to deal with the GUI, you have the wonderful option of simply doing a silent upgrade using the DBUA. This means you won't be prompted for anything—you simply wait for a couple of hours, and if there are no major problems, your database will be automatically upgraded for you. Use the following command for the silent invocation of the DBUA (assuming your database name is `nina`):

```
$ dbua -silent -dbName nina
```

FIGURE I-3

The Welcome Screen of the Database Upgrade Assistant



This command is all you need to migrate your current database to Oracle Database 10g.

## The Upgrade Process with the DBUA

The following are the steps to performing a database upgrade using the DBUA in the interactive GUI mode (you click Next to continue after each step):

- **Select the database instance to upgrade.** In the Selecting a Database Instance screen, first ensure the chosen database has already been started. Then select the database instance to be upgraded.
- **Create the SYSAUX tablespace.** Next is the Sysaux Tablespace screen. Let Oracle create the new SYSAUX tablespace for you.



**The SYSAUX tablespace is a mandatory new tablespace in Oracle Database 10g. Oracle uses the SYSAUX tablespace to store the data for several Oracle features, including the RMAN. If you use the DBUA to upgrade, Oracle will automatically create the SYSAUX tablespace for you. On the other hand, if you manually upgrade, you must create the SYSAUX tablespace, as described in the “Steps in the Manual Upgrade Process” section later in this chapter.**



- **Choose the option to recompile invalid objects.** The next screen gives you the option of automatically recompiling any invalid database objects after the database upgrade is completed. Let Oracle recompile objects that are invalid.

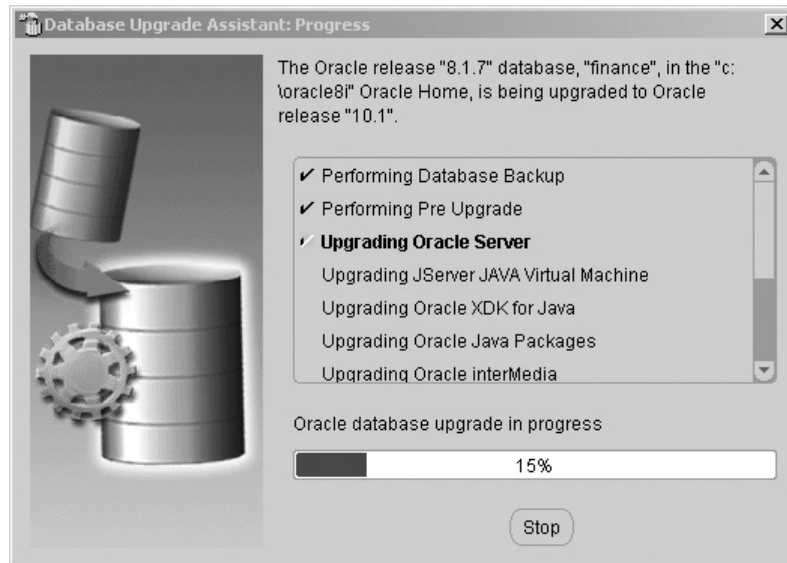
***Letting Oracle recompile all invalid objects at the end of the upgrade is equivalent to manually running the `utlrp.sql` script during a manual upgrade.***

- **Back up your database.** The next screen gives you a last chance to do a cold backup of your database, if you haven't already done so. If you (as any sane DBA would) have already done your backups, choose the I Have Already Backed Up My Database option.
- **Choose the OEM configuration.** The next screen is the Management Options screen, which provides options to configure the OEM, as well as backups. You can choose either the Grid Control or Database Control version of the OEM to manage your upgraded database. As noted previously in this chapter, the Database Control component comes with your database installation software, and Oracle automatically installs it when you install Oracle Database 10g, but you must install the Grid Control from separate software. Grid Control assumes that you have already created a Management Service on one of the servers in your grid. If you haven't already installed the Grid Control software, it may be best to choose the Database Control at this early stage in the game.
- **Define the Flash Recovery Area.** The DBUA then asks you to choose a flash recovery area, which, as described earlier in this chapter, is used as the default area for your backup and recovery-related operations.
- **Choose database passwords.** In the Database Credentials screen, the DBUA asks you to choose passwords for users like SYSMAN and DBSNMP, which are default users of all Oracle databases.
- **Verify the details of the upgrade** The Upgrade Summary screen comes next. The summary includes the source and target database names and database versions. You should verify the following important details at this stage:
  - Database name
  - Oracle Homes for the source and target databases
  - Source and target database versions

The Upgrade Summary screen also shows an estimate of the database upgrade time. Click Finish to start the upgrade process. Figure 1-4 shows the upgrade progress screen of the DBUA.

**FIGURE 1-4**

The DBUA  
upgrade progress  
screen



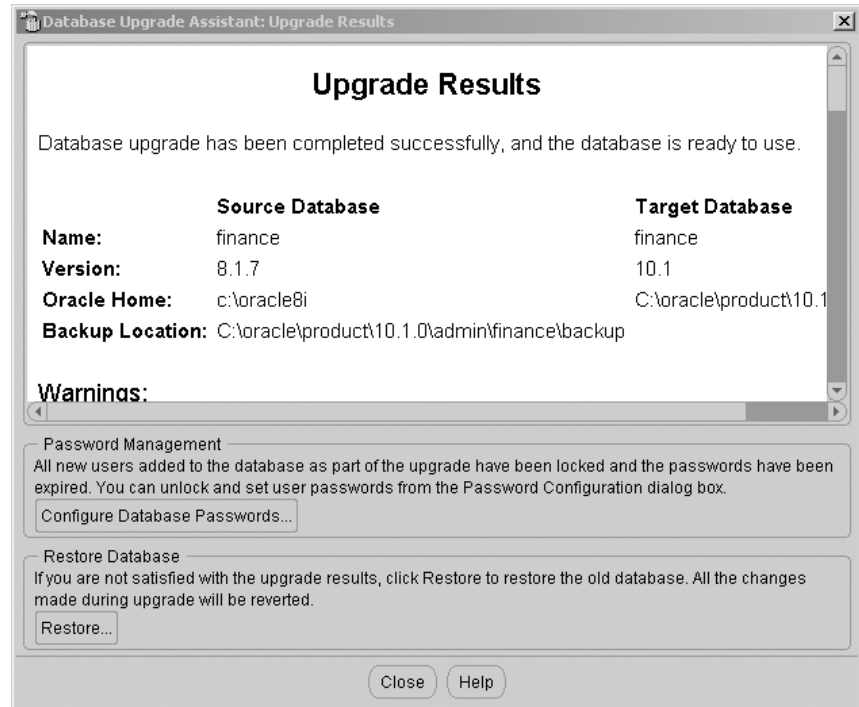
- **Accept upgrade results or restore from backup.** Finally, after the upgrade is completed, the DBUA shows you the results in the Upgrade Results screen, as shown in Figure 1-5. You can do three tasks using the Upgrade Results screen:
  - Check the upgrade details.
  - Manage passwords in the database.
  - If you wish, restore the pre-upgrade database.

## How to Restore to the Pre-Upgrade Version

Amazingly, you can now choose to go back to the pre-upgrade version of your database, if you so choose. You can do this by simply clicking the Restore button. This will restore the database to the pre-upgrade version, if you have chosen the option to have the RMAN perform a cold backup during the DBUA process. If you didn't use the DBUA to perform the backup (through the RMAN), it will tell you that it cannot perform a restore.

**FIGURE 1-5**

The DBUA  
Upgrade Results  
Screen



on the  
job

**You can automatically revert to an older version of your database after the 10g upgrade completes only if you've used the DBUA to back up your database first.**

What if you chose not to restore the database to the pre-upgrade status, and later found that you had made a terrible mistake in turning down the kind offer of the DBUA to undo the upgrade process? Not to worry—you can just run an Oracle-provided script to restore back to the pre-upgrade database. This script is created by Oracle when you choose to back up the database during the beginning part of the upgrade process. Here's how I ran the script on my Windows machine to undo an Oracle Database 10g upgrade (Note that *finance* is the name of my recently upgraded database.)

```
SQL> @C:\oracle\product\10.1.0\admin\finance\backup\FINANCE_restore.bat
```

**CERTIFICATION OBJECTIVE 1.08**

## Starting Up the Database Using a New Mode When Upgrading

Although Oracle strongly recommends that you use the DBUA tool to upgrade your database, we'll step through the manual database process in this section, so you understand the various activities that a database upgrade involves. This will also give you the chance to see how to start up a database using a new mode—*startup upgrade*—when you are upgrading it to Oracle Database 10g.

Remember that the DBUA automatically performs virtually all the same steps for you behind the scenes. After you run your Upgrade Information Tool, as described earlier in this chapter, make sure you've made all the recommended changes, so you don't run into unnecessary error messages during the upgrade process.

### Steps in the Manual Upgrade Process

Here are the steps in the manual upgrade process:

1. Start a spool file.
2. Start the Upgrade Information Tool.
3. Back up your current database.
4. Copy your present `init.ora` file to the new Oracle Database 10g default location.
5. If you are using a Windows-based version of Oracle, remove the old instance.
6. Start up the database.
7. Create the SYSAUX tablespace.
8. Run the upgrade script.
9. Verify the existence of invalid objects.
10. Run the `utlrp.sql` script to automatically recompile and validate all the invalidated PL/SQL and Java code in your database.
11. Run the `utlu101s.sql` script (the Post-Upgrade Status Tool).

These steps are detailed in the following sections:

## Start a Spool File

Start a spool file, so you can easily review the upgrade process.

```
SQL> spool upgrade.log
```

## Run the Upgrade Information Tool

Run the Upgrade Information Tool while connected to the database that you are upgrading. The Upgrade Information Tool is run by executing the *utlu101i.sql* script, which is located in the new Oracle Database 10g software's ORACLE\_HOME/rdbms/admin directory.

```
SQL> @$ORACLE_HOME/rdbms/admin/utlu101i.sql
```

To review the results of this pre-upgrade check, turn spooling off using the following command:

```
SQL> spool off
```

Here's the *partial* output from a test I ran on my Windows server. Note that I am upgrading an Oracle 8.17 database to the Oracle Database 10g version.

```
*****
Database:
-> name: FINANCE
-> version: 8.1.7.0.0
-> compatibility: 8.1.0
WARNING: Database compatibility must be set to 9.2.0 prior to upgrade.
*****
Update Parameters: [Update Oracle Database 10.1 init.ora or spfile]
WARNING: -> "shared_pool_size" needs to be increased to at least "150944944"
*****
Obsolete Parameters: [Update Oracle Database 10.1 init.ora or spfile]
-> "job_queue_interval"
-> "max_enabled_roles"
*****
Components: [The following database components will be upgraded or installed]
-> Oracle Catalog Views          [upgrade]
-> Oracle Packages and Types     [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.
```

## Back Up Your Database

At this point, shut down and back up your current database, by using either the RMAN or by using user-managed backup techniques.

## Copy Your init.ora File

Copy your present init.ora file to the new Oracle Database 10g default location (for example, \$ORACLE\_HOME/dbs under UNIX). Make all the necessary changes in your init.ora parameter file, as per the Upgrade Information Tool's recommendations.

## Completely Remove Any Windows-Based Oracle Instances

If you are using a Windows-based Oracle version, you need to do the following, to make sure that you completely remove the old instance before creating the new Oracle Database 10g instance.

```
C:\> net stop oracleservicefinance
The OracleServiceFINANCE service is stopping
The OracleServiceFINANCE service was stopped successfully.
C:\> oradim -delete -sid finance
Instance deleted.
C:\>
C:\>oradim -new -sid finance -intpwd financel -startmode auto -
pfile c:\oracle\product\10.1.0\Db_1\database\initfinance.ora
Instance created.
C:\>
```

## Start Up the New Database

Start up the database under the new Oracle Database 10g Home, after making sure that the updated init.ora parameter file is in its default location.

Make sure you start up the Oracle Database 10g in the following manner, to start the upgrade process. Note the use of the new `startup upgrade` command. You must use this command, or the upgrade won't start. Using the `startup upgrade` command tells Oracle to automatically modify certain parameters, including initialization parameters that cause errors otherwise (for example, it sets the `job_queue_processes` parameter to zero).

```
C:\> sqlplus /nolog
SQL*Plus: Release 10.1.0.2.0 - Production on Wed Apr 14 11:13:25 2004
Copyright (c) 1982, 2004, Oracle. All rights reserved.
SQL> connect / as sysdba
```



```

Connected to an idle instance.
SQL> startup upgrade
ORACLE instance started.
Total System Global Area  310378496 bytes
Fixed Size                  788268 bytes
Variable Size               309328084 bytes
Database Buffers            0 bytes
Redo Buffers                 262144 bytes
Database mounted.
Database opened.
SQL>

```

## Create the SYSAUX Tablespace

After you start up the database with the `startup upgrade` command, first create the new mandatory SYSAUX tablespace, as recommended by the Upgrade Information Tool. Here's an example:

```

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

```

At this point, the database is technically converted into a Oracle 10g version, although you yet have to run the main upgrade script. The following query demonstrates this:

```

SQL> select * from v$version;
BANNER
-----
Oracle Database 10g Enterprise Edition Release 10.1.0.2.0 - Prod
PL/SQL Release 10.1.0.2.0 - Production
CORE      10.1.0.2.0      Production
TNS for 32-bit Windows: Version 10.1.0.2.0 - Production
NLSRTL Version 10.1.0.2.0 - Production
SQL>

```

## Run the Upgrade Script

Although the database is opened, and although the query on `V$VERSION` shows that your database is now an Oracle 10g version, you still need to run the actual upgrade script, which is your next task. Each version of Oracle has a separate upgrade script. Here are the various versions of Oracle you can directly upgrade to Oracle 10g, and the corresponding upgrade script for each.

- 8.0.6: u0800060.sql
- 8.1.7: u0801070.sql
- 9.0.1: u0900010.sql
- 9.2: u0902000.sql

Since I am upgrading a 8.1.7 database, I choose the u08010870.sql upgrade script, which is located in the \$ORACLE\_HOME/rdbms/admin directory.

```
SQL> @c:\oracle\product\10.1.0\Db_1\rdbms\admin\u0801070.sql
```

The u0801070.sql upgrade script calls several other scripts to manage the upgrade process. The upgrade script will call various Oracle SQL scripts, as shown here:

```
call i0801070.sql:
This loads all tables that are necessary to perform basic sql commands.
call utlip.sql to invalidate views, procedures, packages
call c0801070.sql:
This performs all necessary dictionary upgrades to bring the db from 8.1.7 to
the new release.
call a0801070.sql:
This performs all necessary upgrade using anonymous blocks.
call cmpdbmig.sql
```

This calls the upgrade scripts for all of the components that have been loaded into the database. The script uses procedures from the DBMS\_REGISTRY package to execute various component upgrades.

## Verify the Existence of Invalid Objects

During the upgrade process, Oracle will create, drop, and alter several database tables and other objects. Thus, it's only natural that several internal Oracle packages and procedures will become invalidated during the course of the upgrade. Once the upgrade script finishes, you can verify the existence of invalid objects by running the following script:

```
SQL> select count(*) from dba_objects
       2 where status = 'INVALID';
COUNT(*)
-----
      16742
1 row selected.
SQL>
```

## Recompile and Validate the Invalidated PL/SQL and Java Code

Wow! A large number of procedures and packages have been invalidated, but not to worry! Just run the Oracle `utlpr.sql` script, which will automatically recompile and validate all the invalidated PL/SQL and Java code in your database. Of course, even if you don't recompile all the invalidated packages and procedures in this way, Oracle will validate each object at run time, when a user accesses the object. However, this might lead to a deterioration in run-time performance, so you're better off recompiling all the objects during the upgrade process itself. Note that the `utlpr.sql` script actually calls another script—the `utlprp.sql` script, which is a wrapper based on the `UTL_RECOMP` package.

```
SQL> @C:\oracle\product\10.1.0\Db_1\rdbms\admin\utlpr.sql
TIMESTAMP
-----
COMP_TIMESTAMP UTLRP_BGN  2004-04-14 19:03:37
1 row selected.
PL/SQL procedure successfully completed.
TIMESTAMP
-----
COMP_TIMESTAMP UTLRP_END  2004-04-14 19:39:36
1 row selected.
PL/SQL procedure successfully completed.
PL/SQL procedure successfully completed.
SQL>
```

Just to make sure that there aren't any more invalid objects left, run the following query again:

```
SQL> select count(*) from dba_objects
      2  where status = 'INVALID';
      COUNT(*)
-----
              0
1 row selected.
SQL>
```

Besides validating all remaining invalidated database objects, the `utlpr.sql` script also checks for the validity of each individual component in the database. As you have seen earlier in this chapter, the `DBA_SERVER_REGISTRY` view will contain information about all the component entries. The `utlpr.sql` script will update `DBMS_SERVER_REGISTRY` after it runs, to let Oracle know that it has validated all objects.

**exam**  
**Watch**

**You can also revert back manually to the older database by using the `DB_Name_restore.bat` file (under Windows). To restore your old database using either of the two methods mentioned here requires that you enabled the DBUA to perform a backup of your database (with the RMAN). If you chose not to have the DBUA perform a pre-upgrade backup, you can still revert to your old pre-Oracle 10g version database by simply restoring from your own backups. The key here is to ensure that you have a backup, either made by you or created by the DBCA, before you start the upgrade process.**

### Run the Post-Upgrade Status Tool

Once you complete the upgrade process, run the following script, which works as the Post-Upgrade Status Tool.

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

The Post-Upgrade Status Tool provides a summary of the upgrade process, as shown in the following listing from my upgrade process (Note that the `utlu101s.sql` script is followed by the word `TEXT`, to enable the printing of the following output):

```
SQL> @c:\oracle\product\10.1.0\Db_1\rdbms\admin\utlu101s.sql TEXT
PL/SQL procedure successfully completed.
Oracle Database 10.1 Upgrade Status Tool 13-APR-2004 03:21:22
-> 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 interMedia                       Normal successful completion
-> Spatial                                  Normal successful completion
-> Oracle Text                             Normal successful completion
No problems detected during upgrade
PL/SQL procedure successfully completed.
SQL>
```

Where does the Upgrade Status Tool get its information about the various components? The tool simply queries the `DBA_SERVER_REGISTRY` table to determine the upgrade status of each individual component. You can get the same output as that produced by the `utlu101s.sql` script, by running the following query:

```
SQL> select comp_id, comp_name, version, status
       from dba_server_registry;
```

Based on the outcome of your database upgrade, the Upgrade Status Tool will tell you one of the following: all the database components have been successfully upgraded (Normal Successful Completion), or some of the components haven't been cleanly upgraded. If a component hasn't been cleanly upgraded, the Upgrade Status Tool will tell you the corrective action to take to fix the problem, as shown in the following example.

```
SQL> @C:\oracle\product\10.1.0\Db_1\rdbms\admin\utlu101s.sql TEXT
PL/SQL procedure successfully completed.
Oracle Database 10.1 Upgrade Status Tool 14-APR-2004 04:59:46
-> Oracle Database Catalog Views      Normal successful completion
-> Oracle Database Packages and Types Problem(s) detected
WARNING: --> component status is not valid
--> version is correct
--> check upgrade log file for errors
--> script="re-run base update "u" script"
-> JServer JAVA Virtual Machine      Normal successful completion
-> Oracle XDK                        Normal successful completion
-> Oracle Database Java Packages     Normal successful completion
-> Oracle interMedia                Normal successful completion
-> Spatial                          Normal successful completion
-> Oracle Text                      Normal successful completion
PL/SQL procedure successfully completed.
SQL>
```



***Don't try to start your newly upgraded database under your old Oracle Home for any reason—this will corrupt your database.***

## After the Upgrade

You have now successfully upgraded your database to the Oracle Database 10g (10.1.0.2.0) version. Of course, at this point, all your old application code, as well as database features, are strictly at the Oracle8i or Oracle9i level, depending on what your pre-upgrade database version was. Now, you can check out the various new features of Oracle Database 10g, as well as most of the Oracle database features, see which ones are useful to you. The rest of this book focuses on explaining all the new database administration features in Oracle Database 10g.

## INSIDE THE EXAM

The exam contains questions about tracking Oracle Database 10g feature usage. Pay particular attention to tracking feature usage through OEM as well as the `DBA_FEATURE_USAGE_STATISTICS` view. Something else that the test might focus on are the high-water mark statistics for various database objects. Which database objects does Oracle collect high-water mark statistics for?

The test expects you to know the new Oracle Policy Framework. How do you prioritize policy violations along various categories? The test may include a question about the new Database Cloning Tool (OEM Database Cloning wizard). What are the various features of the source and target databases during a cloning operation?

You must understand the `COMPATIBLE` initialization parameter accurately for the test. The exam tests your knowledge of the DBCA tool in creating a new database. Please look at the various DBCA screens carefully. What management options does the DBCA enable you to configure?

The exam will include a question or two about the upgrade process. You must understand how to use the Pre-Upgrade Information Tool and the Post-Upgrade Status Tool. What scripts start these tools? You must also understand the role of the `DBA_SERVER_REGISTRY` during a database upgrade. How do you recompile invalid objects?

## CERTIFICATION SUMMARY

This chapter provided you with a summary of Oracle 10g's innovations in the software installation and database upgrade areas. First, you looked at the support of new installation features in Oracle 10g. These included the changes to the Oracle Enterprise Manager (OEM) tool. You learned the difference between local and central installation of the OEM. Other installation new features include the file storage options and backup and recovery options. You also learned about the changes in the response file installation, as well as the new irreversibility factor relating to the `COMPATIBLE` initialization parameter.

You saw how you now perform an Oracle database installation from a single CD, with minimal hardware requirements. You then looked at how Oracle 10g provides you both pre-installation and post-installation support tools to ensure easy software installation. Next, you learned about the new easier and cleaner Oracle software deinstallation process.

The Database Creation Assistant (DBCA) has many changes, and this chapter reviewed all of them. These include the mandatory creation of the SYSAUX tablespace, the creation of the flash recovery area, and the Automatic Storage Management (ASM) storage option. Management options include a choice between the OEM Database Control and the OEM Grid Control. You also saw how the OEM can now be used to link with Oracle*MetaLink* service, perform database cloning, and enforce a predetermined policy-based configuration framework.

An interesting new feature is how you can track database feature usage and high-water mark (HWM) statistics. This chapter demonstrated how to use this feature both manually as well as through the OEM Database Control.

In the database upgrade section, you first reviewed the supported upgrade paths to the new Oracle Database 10g version. I then took you step-by-step through both a manual and a DBUA database upgrade. You learned about the Oracle pre-upgrade checks and the post-upgrade validation as well.



## TWO-MINUTE DRILL

### Installation New Features Support

- You have more management choices than with previous versions if you decide to let Oracle create the starter database.
- The Oracle Universal Installer (OUI) now performs a series of pre-installation checks before the installation process can start.
- You have a choice between two types of Oracle Enterprise Manager (OEM): local management using the OEM Database Control or central management using the OEM Grid Control. With the Database Control, you can manage only a single database. With the Grid Control, you can manage several nodes at once.
- The OEM Database Control comes with the Oracle 10g software installation. You need to install the OEM Grid Control separately, using a separate CD.
- The default port number for the OEM Database Control is 5500.
- The installer enables you to configure default backup strategies.
- The installer gives you three storage choices: traditional UNIX/Windows file systems, Automatic Storage Management (ASM), and raw devices.
- You need to specify database user (SYS and SYSTEM, for example) passwords during database creation time.
- You can invoke the OUI from any node on a cluster that is part of an installation.
- The Oracle 10g installer supports the installation of Cluster Ready Services (CRS).
- You can use the OEM Grid Control to perform enterprise-wide configuration changes.
- There are several changes in the response file creation, enabling a truly “silent” installation. These changes include the following:
  - New header formats
  - No need to specify X server
  - No need to specify the DISPLAY variable
  - No instantiation of GUI classes



## Performance Enhancements to the Installation Process

- ❑ Performance enhancements include a single CD Oracle software installation. Companion CDs contain documentation, client software, demos, and the OEM Grid Control components.
- ❑ Minimal hardware requirements are now necessary for Oracle 10g installation.
- ❑ The OUI performs most of the automatic pre-installation checks. It checks for the correct operating system version, kernel parameters, patches, disk space, and so on.
- ❑ Oracle also provides a new post-installation tool called the Upgrade Status Tool. You can invoke the Upgrade Status Tool by running the script `$ORACLE_HOME/rdbms/admin/utlu101s.sql`.
- ❑ Oracle 10g enables a cleaner software deinstallation process, wherein it removes all the necessary files automatically as well as performing the necessary Registry changes in Windows systems.
- ❑ Due to the new irreversible file compatibility feature, you cannot revert to a lower compatibility value after you create a new Oracle 10g database.
- ❑ The minimum value of the compatibility setting is 9.2.0.
- ❑ The default setting of the compatibility setting is 10.0.0. This is also the maximum value for the `COMPATIBLE` initialization parameter under Oracle Database 10g Release 10.1.
- ❑ You now need a new mandatory tablespace, called the `SYSAUX` tablespace, to store database metadata relating to various tools like the `LOGMNR`. This brings the number of minimum tablespaces in an Oracle database to two; `SYSTEM` is the other mandatory tablespace.
- ❑ The Database Configuration Assistant (DBCA) automatically creates the `SYSAUX` tablespace for you.
- ❑ The flash recovery area is space that is dedicated to all backup and recovery-related activities and files. The DBCA automatically creates this for you.
- ❑ You can now link OEM directly to the `OracleMetaLink` service. This enables the automatic tracking of new software patches.
- ❑ You can now use the OEM to clone databases, provided you use `RMAN` to back up the source databases. You can do this by selecting `Maintenance | Deployments | Clone Database` from the OEM home page.

### **Simplified Instance Configuration**

- To simplify basic instance configuration, Oracle now divides initialization parameters into two groups: basic and advanced.
- There are about 25 to 30 basic initialization parameters (I saw the precise figure of 28 initialization parameters in one Oracle Corporation white paper).
- There are several enhancements in the DBCA, which enables you to configure the following:
  - SYSAUX tablespace
  - Flash recovery area
  - Automatic Storage Management (ASM)
- You can now link the OEM directly to the *OracleMetaLink* service.
- You can automatically download patches through *OracleMetaLink* now.
- Using the OEM, it is easy to clone databases now.

### **Viewing Database Feature Usage Statistics**

- You can now view database feature usage statistics for various Oracle features like auditing through the OEM Database Control. You can also view the feature usage by querying the view `DBA_FEATURE_USAGE_STATISTICS`.
- You can view high-water mark (HWM) statistics for objects like tables, indexes, and datafiles. The new view that you can use to check HWM statistics is called `DBA_HIGH_WATER_MARK_STATISTICS`.
- You can now track whether your database is following predetermined database configuration policies, and record all policy violations. Again, you can use the OEM to perform this task.

### **Supported Upgrade Paths to Oracle Database 10g**

- You can directly upgrade to the Oracle 10g (10.1.0) version only if you are currently using the 8.0.6, 8.1.7, 9.0.1, or a 9.0.2 version database.
- You can upgrade to Oracle 10g by using the Database Upgrade Assistant (DBUA) or by using a manual method.

## Using New Utility to Perform Pre-Upgrade Validation Checks

- Use the new Upgrade Information Tool to gather pre-upgrade information. You invoke this tool by running the `utl101i.sql` script located in the `$ORACLE_HOME/rdbms/admin` directory.

## Using the Simplified Upgrade Process

- During a database upgrade, Oracle will now consult a new internal table called the `DBA_SERVER_REGISTRY`. This table contains names of all the components you are going to be upgrading, as well as their post-upgrade status.
- Oracle recommends that you use the DBUA to perform all database upgrades, in order to reduce mistakes and make the process a lot simpler.
- The DBUA performs pre-upgrade checks including the following:
  - Changes the `ORACLE_HOME` settings
  - Changes the `/etc/oratab` entries
  - Automatically deletes all obsolete initialization parameters
  - Checks for adequate physical space for the upgrade
  - Checks for any desupported character sets, invalid data types, and so on
- The DBUA will also provide you with a choice to back up your old database.
- Following are the upgrade steps using the DBUA:
  - Start the DBUA by typing the command `dbua` in UNIX/Linux systems. In Windows systems, click the Database Upgrade Assistant button under the Oracle Programs list.
  - Select the database to be upgraded from the list provided by the DBUA.
  - Allow the DBUA to create the mandatory `SYSAUX` tablespace.
  - Choose automatic recompiling of all objects after the upgrade process.
  - Choose either the OEM Database Control or Grid Control under management options.
  - Choose a flash recovery area, where Oracle will store all backup and recovery-related files.
  - Approve the upgrade process after looking at the summary provided by the DBUA.

- If you choose to revert to the pre-upgrade version after the upgrade process, you can do so by clicking the Restore button in the final DBUA Upgrade Results screen.

### **Starting Up the Database Using a New Mode When Upgrading**

- Following is a summary of the steps in a manual upgrade process:
  - Adjust the tablespace and redo log sizes if necessary.
  - Remove any deprecated or obsolete parameters and add any necessary new parameters.
  - Back up your current database.
  - Copy the present init.ora file after you modify it, to the new Oracle 10g default init.ora location.
  - Under a Windows system, make sure the existing instance is stopped and deleted using the `oradim` utility. Create a new instance using the same utility.
  - Start the database from the new Oracle 10g Home.
  - Begin the upgrade process by issuing the command `startup upgrade`.
  - Create the SYSAUX tablespace.
  - Run the appropriate upgrade script for your database version (for example, `u0801070.sql` for an 8.1.7 version database).
  - Once the upgrade script finishes running, compile any invalid objects by running the `utlrp.sql` script.
  - Check to make sure there aren't any invalid objects.
  - Invoke the Upgrade Status Tool by running the `utlu101s.sql` script from the `$ORACLE_HOME/rdbms/admin` directory. Remember that you can get this information by querying the `DBA_SERVER_REGISTRY` table directly.
  - If the Upgrade Status Tool indicates “Normal Successful Completion” for all the database components, your upgrade is successful.
  - For a successful upgrade, the query on the `DBA_SERVER_REGISTRY` view should indicate “valid” under the status column for all the database components in the new Oracle Database 10g.

## SELF TEST

The following questions will help you measure your understanding of the material presented in this chapter. Read all the choices carefully because there might be more than one correct answer. Choose all correct answers for each question.

### Installation New Features Support

1. Which of the following sets of tablespaces is mandatory for any Oracle 10g database?
  - A. SYSTEM, SYSAUX, and UNDOTBS
  - B. SYSTEM and UNDOTBS
  - C. SYSAUX and UNDOTBS
  - D. SYSTEM and SYSAUX
2. What does a DBA need to do for Oracle to perform the automatic pre-installation requirements check, when you start installing the Oracle Database 10g Server software?
  - A. Run the Upgrade Information Tool manually.
  - B. Do nothing—Oracle will automatically perform the pre-installation checks.
  - C. Use the `upgrade database` command to open the database.
  - D. Execute the `utls10x.sql` script.
3. What does the Enterprise Configuration Management Tool do?
  - A. Helps you configure enterprise-wide configuration through the OEM Grid Control
  - B. Helps you configure enterprise-wide configuration through the OEM Database Control
  - C. Helps you configure the Oracle Database Control tool
  - D. Collects only database information, not host information

### Performance Enhancements to the Installation Process

4. During a response file-based “silent” Oracle installation, what do you need to do?
  - A. Specify an X server while performing installations in a character mode console.
  - B. You don’t need to specify an X server while performing database installation in a character mode console.
  - C. Set a `DISPLAY` variable on UNIX systems.
  - D. Use no header formats in the response file.

## 54 Chapter 1: Installation, Server Configuration, and Database Upgrades

5. When you deinstall a Windows Oracle server installation, which of the following is true?
  - A. Oracle will automatically clean up the Windows Registry for you.
  - B. You must manually clean up the Windows Registry.
  - C. The Windows Registry doesn't need to be changed.
  - D. The Windows operating system automatically cleans up the Windows Registry for you.
6. When you complete the Oracle software installation, which of the following products does Oracle launch for you?
  - A. OEM Management Agent
  - B. OEM Database Control
  - C. OEM Grid Control
  - D. OEM Management Agent and OEM Database Control

### Simplified Instance Configuration

7. Which one of the following initialization parameters would belong to the basic group of parameters?
  - A. SHARED\_POOL
  - B. CURSOR\_SPACE\_FOR\_TIME
  - C. DB\_NAME
  - D. CURSOR\_SHARING
8. If you use the DBCA to create your new database, when is the creation of the new flash recovery area mandatory?
  - A. Always
  - B. Only if you configure automatic backup using the OEM
  - C. Only if you configure user-managed backups
  - D. Only if you run your database in the archive log mode
9. Which of the following is a new initialization parameter for Oracle Database 10g?
  - A. UNDO\_SUPPRESS\_ERRORS
  - B. PARALLEL\_AUTOMATIC\_TUNING
  - C. RESUMABLE\_TIMEOUT
  - D. SHARED\_SERVER

## Viewing Database Feature Usage Statistics

10. Where are database usage and high-water mark statistics stored?
- A. Flash recovery area
  - B. Automatic Workload Repository
  - C. RMAN Repository
  - D. In the bit map area of the datafiles
11. Which two of the following are examples of an Oracle database high-water mark statistic?
- A. SGA size
  - B. Number of named users
  - C. Size of the largest data segment
  - D. Maximum number of concurrent sessions
12. You have just upgraded your 8.1.7 release Oracle database to the new 10.1.0 version. You did not specify a value for the `COMPATIBLE` initialization parameter. After using the new database for a while, you decide you need to revert to your 8.1.7 database. Which one of the following is true under these circumstances?
- A. You cannot return to the 8.1.7 version.
  - B. Just add `COMPATIBLE=8.1.7` to your initialization parameters. You should then be able to go back to the 8.1.7 version.
  - C. Use the `ALTER DATABASE RESET COMPATIBILITY` command to go back to the 8.1.7 version.
  - D. You cannot go back to a time before the upgrade, unless you do a point-in-time recovery.

## Supported Upgrade Paths to Oracle Database 10g

13. Which of the following Oracle versions is not an approved version for a direct upgrade to Oracle 10g (10.1.0)?
- A. 8.0.6
  - B. 8.1.7
  - C. 9.0.1
  - D. 8.1.6
14. In which of the following modes can you upgrade your current database to the Oracle Database 10g version?
- A. Database Upgrade Assistant
  - B. Oracle Data Migration Assistant

- C. Export and import
- D. Manual upgrade

### **Using New Utility to Perform Pre-Upgrade Validation Checks**

- 15.** What is the name of the Oracle-supplied script that runs the pre-upgrade validation checks?
- A. utlu101i.sql
  - B. utl101i.sql
  - C. utlu101x.sql
  - D. utlu101s.sql
- 16.** If you are using the DBCA to upgrade your database to the Oracle Database 10g version, which of the following is true in relation to the Upgrade Information Tool?
- A. You don't need to run the utlu101u.sql script.
  - B. You must run the utl101u.sql script.
  - C. You must run the utlu101s.sql script.
  - D. You must choose the Pre-Upgrade Information Tool option while using the DBCA.
- 17.** The Upgrade Information Tool provides information about which of the following?
- A. Optimal SGA allocation to the various components of SGA, for example, the shared pool
  - B. Optimal performance features of Oracle Database 10g
  - C. Recommendations for additional space for tablespaces
  - D. A time estimate for the upgrade

### **Using the Simplified Upgrade Process**

- 18.** You are using the DBUA to upgrade your database. One of the steps during the use of the DBUA is a choice regarding the recompilation of invalid objects. Telling the DBUA to recompile all invalid database objects after the upgrade is the same as running which script?
- A. utlrpt.sql
  - B. utlu101i.sql
  - C. utlu101x.sql
  - D. utlrp.sql
- 19.** You have just finished an Oracle Database 10g upgrade using the DBUA tool. You have chosen most of the default settings, including letting the DBUA perform a pre-upgrade backup for you. You are at the very end of the process, when the Upgrade Summary screen shows you the



upgrade details. You suddenly realize that you made a mistake, and you would like to revert to the previous software version, which is an Oracle 8.1.7.3 database. What can you do?

- A. Just start up the new database from the old Oracle 8.1.7.3 Home.
- B. Just click the Restore button on the update summary screen of the DBUA.
- C. You need to start fresh by manually restoring your 8.1.7.3 database.
- D. There is no way to restore to 8.1.7.3 at this point.

### **Starting Up the Database Using a New Mode When Upgrading**

- 20.** When you start up your database just prior to running the upgrade script, you need to use which of the following commands?
- A. `startup mount`
  - B. `startup nomount`
  - C. `startup open`
  - D. `startup upgrade`
- 21.** Once you upgrade your database to the Oracle Database 10g version, which of the following is true?
- A. You can always start it either under the old Oracle Home or the new Oracle Home.
  - B. You will corrupt the database if you try to start it under the old Oracle Home.
  - C. You can start it under the old Oracle Home, as long as your compatibility is set to the old database version level.
  - D. You can start it in either Oracle Home, as long as the compatibility level is set to 10.0.0.

## **LAB QUESTIONS**

### **Lab 1**

Using the OEM Database Control, find the locations of the following:

- Tablespaces
- Flash recovery area

### **Lab 2**

Using the OEM Database Control, how do you find out the initialization parameters for the instance?

**Lab 3**

Manually create a SYSAUX tablespace with a size of 1GB.

**Lab 4**

Show all the command-line steps necessary to access the OEM Database Control (assume that no OEM-related process is currently running on your server).

**Lab 5**

Show the steps necessary to clone a database, and explain the cloning process in detail.

# SELF TEST ANSWERS

## Installation New Features Support

- D.** SYSTEM and SYSAUX are the two mandatory tablespaces, without which you cannot create an Oracle Database 10g database.  
 **A, B, and C** are incorrect, since they include UNDOTBS, which isn't a mandatory tablespace in Oracle Database 10g. Remember that you still can use manually configured rollback segments, in which case you won't need an undo tablespace.
- B.** Oracle will automatically perform the pre-installation checks for you when you start installing the software. There are no scripts to run.  
 **A** is wrong because the Upgrade Information Tool is used only for upgrades, not for installations. It is the same case with option **C** as well. **D** is wrong since the utls101x.sql script performs a post-upgrade check, not a pre-installation check.
- A.** The Enterprise Configuration Management Tool, as its name indicates, helps you make enterprise-wide configuration changes, using the OEM Grid Control.  
 **B** is wrong since, by definition, you cannot perform enterprise-wide changes using the OEM Database Control. **C** is wrong because the Oracle Database Control tool is configured when you install the database software. **D** is wrong since the Enterprise Configuration Tool collects database and host information across the enterprise.

## Performance Enhancements to the Installation Process

- B.** Oracle Database 10g makes the response file-based "silent" installations easier, and one of the ways is by not forcing you to specify an X server in a character mode console.  
 **A** and **C** are incorrect because they are not true in the Oracle 10g version. **D** is wrong, as you still need header formats in the response file, although the newer header formats are a lot easier to edit.
- A.** The Oracle Universal Installer will automatically clean up the Windows Registry for you as part of the software deinstallation process.  
 **B** is wrong because you don't need to perform a manual cleanup of the Windows Registry anymore in the Oracle 10g database. **C** is incorrect since the Registry changes are necessary after a software deinstallation. **D** is wrong since the Windows operating system doesn't automatically clean up the Registry after an Oracle deinstallation.
- A, B, and D.** Oracle launches both the OEM Management Agent and the OEM Database Control automatically after installation.  
 **C** is wrong because the Oracle Universal Installer never launches the OEM Grid Control as part of Oracle server installation.

## Simplified Instance Configuration

7.  **A** and **C**. Both `SHARED_POOL` and `DB_NAME` belong to the basic set of initialization parameters, which should be sufficient in most cases. The parameters in the advanced list are necessary only under some special conditions.
- You can easily see that **B** and **D** refer to initialization parameters that belong to the advanced group, not the basic group of parameters.
8.  **B**. The creation of the flash recovery area is mandatory only if you configure automatic backups using the OEM.
- A** is clearly wrong, since the flash recovery area is not mandatory under all circumstances. **C** is wrong because configuring user-managed backups doesn't have anything to do with the flash recovery area. **D** is wrong as well, since the archive log mode in which the database is running has no bearing on the flash recovery area.
9.  **C**. `RESUMABLE_TIMEOUT` is a new Oracle Database 10g initialization parameter.
- A**, **B**, and **D** refer to initialization parameters that have become obsolete in Oracle Database 10g.

## Viewing Database Feature Usage Statistics

10.  **B**. The MMON process periodically collects database usage and high-water mark statistics and stores them in the Automatic Workload Repository.
- A** is wrong since the flash recovery area is free space reserved for recovery-related activities. **C** is wrong since the RMAN repository is used to store backup-related data. **D** is clearly wrong, as bit maps in datafiles are used mostly for storage-related purposes.
11.  **C** and **D**. The size of the largest data segment shows the “high point” of resource usage for that data segment, and thus it captures a high-water mark statistic. The same is true of the maximum number of concurrent sessions statistics.
- A** and **B** refer to statistics that aren't part of the high-water mark statistics attributes.
12.  **D**. This is a tough question. Once you used the database for a while, you cannot set the compatibility back to a lower version. The compatibility level can be set only higher, not lower. You just can't start the new database with a lower level of compatibility. If you are prepared to lose all the changes made since the upgrade, of course, you can always perform a point-in-time recovery to a time that precedes the upgrade.
- A** is wrong since it implies there is no way you can go back to the previous version of the software. **B** is wrong since you cannot set compatibility to a lower level compared to the existing level. **C** is wrong since the `alter database reset compatibility` command is obsolete in Oracle Database 10g.

## Supported Upgrade Paths to Oracle Database 10g

13.  **A, B, and C.** These answers all represent database versions that permit a direct upgrade to Oracle Database 10g.  
 **D.** 8.1.6 belongs in the group of database versions that you cannot directly upgrade to Oracle Database 10g.
14.  **A, C, and D.** You can use either the DBUA or manual upgrade procedure to perform an identical upgrade. Export and import also enable an upgrade to Oracle Database 10g.  
 **B.** The Oracle Data Migration Assistant is the predecessor to the DBUA in previous versions of Oracle databases.

## Using New Utility to Perform Pre-Upgrade Validation Checks

15.  **A.** The utlu101i.sql script located in the \$ORACLE\_HOME/rdbms/admin directory is the correct script to run the Upgrade Information Tool.  
 **B** misspells the correct script name. **C** refers to the wrong script. **D** is wrong because you use this script for post-upgrade verification.
16.  **A.** It is true that if you use the DBCA, you don't need to run the utlu101i.sql manually. The DBCA runs this for you, so it can provide you with vital upgrade-related information.  
 **B** is wrong, since you don't need to run any scripts yourself, if you are using the DBCA. The same reasoning applies to choice **C**. **D** is wrong because the Upgrade Information Tool isn't presented as an option to you by the DBCA; it automatically uses the tool before each database upgrade, by invoking the utlu101i.sql script.
17.  **C and D.** Both recommendations about additional space for tablespaces and upgrade time estimates are part of the Upgrade Information Tool information.  
 **A and B** are invalid answers since the Upgrade Information Tool doesn't give you any performance-related information. Its goal is to ensure that you can perform the upgrade successfully.

## Using the Simplified Upgrade Process

18.  **D.** The utlrp.sql script will recompile all invalid objects in your database.  
 **A** misspells the correct answer. **B** and **C** both refer to pre-upgrade information scripts, not scripts that you run after an upgrade.
19.  **B.** As long as you have backed up the database using the DBUA, you can easily revert to the pre-upgrade version of the database by clicking the Restore button after the upgrade is completed.  
 **A** is wrong because you'll end up corrupting your database if you start the upgraded database from its old Oracle Home location. **C** is wrong because you don't need to manually

restore the database—as long as you let it perform a backup beforehand, the DBUA will perform the restore for you. **D** is incorrect because you can restore to 8.1.7 either through the DBA or through a manual restore, if necessary.

### Starting Up the Database Using a New Mode When Upgrading

- 20.**  **D.** You start a database with the new `startup upgrade` command prior to running your upgrade scripts. Oracle will issue an error otherwise!  
 **A, B, and C** are incorrect, since they don't use the keyword `upgrade`.
- 21.**  **B.** Once you upgrade your database to the Oracle Database 10g version, you must start it under the new Oracle Database 10g Home. If you start it up under the old Oracle Home, you may corrupt the database.  
 **A and D** are incorrect, since you cannot start the upgraded database from its old Oracle Home. **C** is wrong since there is now no way (in Oracle 10g) to go to a lower level of database compatibility.

## LAB ANSWERS

### Lab 1

To find out which tablespaces exist in your database:

1. From the Database Control home page, click the Administration link.
2. From the Administration page, click Tablespaces.

To find out the size and location of the flash recovery area:

1. From the Database Control home page, click the Maintenance link.
2. From the Maintenance page, click Configure Recovery Settings.
3. Go to the flash recovery area settings.

### Lab 2

To find the initialization parameters for the instance:

1. Go to the Database Control home page.
2. Click the Administration link.
3. Click the Initialization Parameters link in the Instance section.

### Lab 3

To manually create a SYSAUX tablespace with a size of 1GB:

```
SQL> CREATE TABLESPACE sysaux DATAFILE 'sysaux01.dbf'
      SIZE 1000M REUSE
      EXTENT MANAGEMENT LOCAL
      SEGMENT SPACE MANAGEMENT AUTO
      ONLINE;
```

### Lab 4

The command-line steps to access the OEM Database Control are as follows:

1. Check whether the `dbconsole` process is running. (Make sure your database is up as well. If it isn't, there isn't anything to manage!)
2. If the `dbconsole` process isn't running, start it up by using the following command:

```
$> emctl start dbconsole
```

3. Once the `dbconsole` process starts up, you can access the Database Control using the following URL in your web browser:

```
http://hostname:5500/em
```

5500 is the default HTTP port number, and it may be different in your case. Make sure you log in as the SYS user or some other user with SYSDBA privileges.

### Lab 5

You must use RMAN to perform backups. You can use the Clone Database wizard (also referred to as the Clone Database Tool) to create a clone of an existing database. Following is a brief summary of the cloning process:

- The source database must be kept open.
- The source database must belong to an Oracle 8.1.7 or higher version.
- The Clone Database Tool (Clone Database wizard) will back up the source database files and copy them to the target location.
- The backup files are used to start up the new instance (in the OPEN mode), which has the same initialization files as the source database.

Using the OEM, this is how you clone a database:

1. From the Database Control home page, click the Maintenance link.
2. Click the Clone Database link under the Deployments section.