

EXAM OBJECTIVE: SIMPLIFY INSTANCE CONFIGURATION USING A SUBSET OF INITIALIZATION PARAMETERS

In Oracle 10g, there is a 90% reduction of initialization parameters; there are less than 30 basic parameters that need to be configured. Only a few parameters need to be set explicitly as the default values for the rest of them are adequate in a majority of cases.

In Oracle 10g, parameters have been categorized into basic and advanced parameters. Administrators will be able to restrict their day-to-day interaction with less than 30 basic parameters. The important ones being:

- Compatible
- Processes
- Sessions
- Pga_aggregate_target
- Nls_language
- Nls_territory
- Db_domain
- Shared_servers
- Instance_number
- Cluster_database
- Db_block_size
- Sga_Target
- Control_files
- Db_name
- Db_recovery_file_dest
- Remote_listener
- Db_recovery_file_dest_size
- Db_create_online_log_dest_n
- Db_create_File_des
- Log_archive_Dest_n
- Log_archive_dest_state_n
- Remote_login_passwordfile
- Db_unique_name

The COMPATIBLE Initialization Parameter

You can control the compatibility of your database with the COMPATIBLE initialization parameter. For the Oracle 10g database it defaults to 10.0.0. This permits the use of the new features of Oracle 10g to be made available.

Before you can upgrade to the Oracle 10g release, you need to set the COMPATIBLE initialization parameter to 9.9.0. This is the lowest possible setting for the Oracle Database 10g release 10.1. In this case only a subset of the 10g features will be made available.

After upgrading to the new Oracle 10g release, the COMPATIBLE parameter can be set to match the release number of the new release. This will permit you to use all the new features of the Oracle 10g database. The downside in doing this is that you cannot downgrade back to a previous release. If you want to downgrade, you must leave the COMPATIBLE initialization parameter set to 9.2.0 after the upgrade.

You can view the current value of the COMPATIBLE initialization parameter, by issuing :

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

EXAM OBJECTIVE: USE DBCA TO CLONE THE DATABASE

Cloning helps the customer to duplicate well-tested and certified environments to other nodes so that systems behave predictably and eliminate runtime surprises. Oracle Enterprise Manager 10g Grid Control – Oracle's central management framework – provides users with a convenient and flexible way to intelligently duplicate Oracle software installations (a.k.a. Oracle homes) across hosts.

It also save time and effort since cloning eliminates the lengthy and tedious processes to install, upgrade and patch to reach a desired level of software or the database.

Prior to oracle 10g, cloning was a manual process that involved error prone and tedious steps. The DBA also had to account for platform specific variations and make manual changes to make the cloned environment "supportable" over the future. The wizard driven cloning mechanism in Enterprise Manager 10g allows customers to:

- Build test environments identical with the production environment
- Create multiple clones from a single source using the "multicasting" capability, reducing human effort to the minimum.
- Cloning operations can be scheduled as Enterprise Manager jobs to be run during off-hours in order to minimize network load.
- Additionally, users can create a library of master installations. Frequently, these master installations have been tested extensively and are certified for use in development, testing or production. Installations that are stored in

this library can be used repeatedly in cloning operations eliminating the need for maintaining actual Oracle homes and tying up valuable host capacity.

DBCA templates are XML files that contain information required to create a database. Templates are used in DBCA to create new databases and make clones of existing databases. The information in templates includes database options, initialization parameters, and storage attributes (for datafiles, tablespaces, control files and redo logs).

Templates can be used just like scripts, and they can be used in silent mode. But they are more powerful than scripts, because you have the option of cloning a database. This saves time in database creation, because copying an already created seed database's files to the correct locations takes less time than creating them as new.

Templates are stored in the following directory:

`$ORACLE_HOME/assistants/dbca/templates`

The files in your OH\assistants\dbca\template directory have the extensions.

- **.DBC – Database Template Definition.**
- **.DBF – Database datafiles**

Database Cloning can be performed from the Enterprise Manager Console.

EXAM OBJECTIVE VIEW DATABASE USAGE STATISTICS THROUGH ENTERPRISE MANAGER

There are two important views in Oracle 10g that display the usage pattern of the database. The **DBA_HIGH_WATER_MARK_STATISTICS**, shows the maximum value of each of the features used in the present database. Look at the example output displayed below:

NAME	HIGHWATER	LAST_VALUE	DESCRIPTION
USER_TABLES	401	401	Number of User Tables
SEGMENT_SIZE	1237319680	1237319680	Size of Largest Segment (Bytes)
PART_TABLES	12	0	Maximum Number of Partitions belonging to an User Table
PART_INDEXES	12	0	Maximum Number of Partitions belonging to an User Index
USER_INDEXES	832	832	Number of User Indexes

SESSIONS	19	17	Maximum Number of Concurrent Sessions seen in the database
DB_SIZE	7940079616	7940079616	Maximum Size of the Database (Bytes)
DATAFILES	6	6	Maximum Number of Datafiles
TABLESPACES	7	7	Maximum Number of Tablespaces
CPU_COUNT	4	4	Maximum Number of CPUs
QUERY_LENGTH	1176	1176	Maximum Query Length

In the example above you can notice several valuable pieces of information on the usage of the database – such as the fact that the users created a maximum of 12 partitioned tables but none are being used now (LAST_VALUE=0). This information is persistent across shutdowns.

The **DBA_FEATURE_USAGE_STATISTICS** provides certain additional information on the usage of features. For example, if you were to find out the last time a certain feature was used this query could be used. Here is how the view looks for the partitioning feature with columns shown in a vertical format:

DBID	: 4133493568
NAME	: Partitioning
VERSION	: 10.1.0.1.0
DETECTED_USAGES	: 12
TOTAL_SAMPLES	: 12
CURRENTLY_USED	: FALSE
FIRST_USAGE_DATE	: 12-nov-2003 10:20:00
LAST_USAGE_DATE	: 10-may-2004 11:49:50
AUX_COUNT	:
FEATURE_INFO	:
LAST_SAMPLE_DATE	: 12-jun-2004 21:21:50
LAST_SAMPLE_PERIOD	: 615836
SAMPLE_INTERVAL	: 604800
DESCRIPTION	: Oracle Partitioning option is being used - there is at least one partitioned object created.

In the output displayed above, the partitioning feature is not being used in the database currently. (CURRENTLY_USED is FALSE). The last time the feature was accessed was 10-may-2004 11:49:50. The usage sampling is done every 604,800 seconds or 7 days, as shown in the SAMPLE_INTERVAL column. The column LAST_SAMPLE_DATE shows the last time this usage was sampled. This indicates how current the information is.

In addition to the above queries, the database usage statistics can also be displayed from the Enterprise Manager Console (EM). **Select Database Control Home**

page -> Administration Tab -> Configuration Management Section -> Database Usage statistics.