

TEMA 02  
Ejercicio práctico 04  
Administración de parámetros

NOMBRE:

GRUPO:

FECHA DE ENTREGA:

CALIFICACION:

### 1.1. OBJETIVO

Comprender y poner en práctica los conceptos asociados con la configuración de los parámetros de una base de datos, en particular, los 3 niveles de aplicación: nivel sesión, nivel instancia y nivel SPFILE, así como las diferentes opciones que existen para obtener y reconstruir tanto PFILEs como SPFILEs.

### 1.2. CREACIÓN DEL DIRECTORIO DE TRABAJO.

Crear un Shell script `s-00-crea-directorios.sh` que realice las siguientes acciones:

- Crear un directorio `/unam-bda/ejercicios-practicos/t0204`
- El directorio deberá pertenecerle al usuario ordinario, pero el grupo asociado a la carpeta deberá ser `oinstall`. De esta forma tanto el usuario `oracle` como el usuario ordinario podrán realizar lecturas y escrituras sobre la carpeta `t0204`. Se empleará el nombre `t0204` dentro de este documento para hacer referencia a este directorio. La carpeta deberá tener todos los permisos para el dueño y el grupo, y permisos solo de lectura para otros usuarios.
- **C1. Incluir en el reporte** el contenido de este script

### 1.3. PARÁMETROS DE LA BASE DE DATOS.

#### 1.3.1. Obtención de los parámetros de la base de datos configurados en el SPFILE.

En esta sección se revisan las opciones existentes para recuperar los parámetros de una instancia que fueron configurados en el SPFILE.

- Opción 1. A través de una bitácora (trace log). No se requiere contar con la instancia iniciada.
- Opción 2. A través de la instrucción `create pfile`. No se requiere contar con la instancia iniciada.
- Opción 3. A través de la vista `v$spparameter`, se requiere de una instancia iniciada.

Es importante conocer y poner en práctica estas opciones ya que pueden aplicarse en caso de pérdida o daño irreparable del SPFILE.

##### 1.3.1.1. Obtención de los parámetros a través del alert log.

Con base a lo revisado en el tema 2, abrir el archivo `alert_<ORACLE_SID>.log`. Ubicar la última ocurrencia de un texto similar al siguiente:

```
Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
Version 18.3.0.0.0.
ORACLE_HOME:      /u01/app/oracle/product/18.0.0/dbhome_1
System name:      Linux
Node name:        jrc-ora-pc.fi.unam
Release:          4.14.35-1902.8.4.el7uek.x86_64
Version:          #2 SMP Mon Dec 9 11:39:31 PST 2019
Machine:          x86_64
Using parameter settings in server-side spfile /u01/app/oracle/product/18.0.0/dbhome_1/dbs/spfilejrcbda2.ora
System parameters with non-default values:
```

*#Aquí se muestra la lista de los parámetros y sus valores en el SPFILE.*

1. Revisar el archivo, incluir el texto similar al anterior y la lista de los parámetros en un archivo de texto `e-01-spparameter-alert-log.txt`. El archivo deberá ser almacenado en la carpeta `t0204`. No se requiere automatizar esta tarea. **C2. Incluir en el reporte** el contenido de este script.

##### 1.3.1.2. Obtención de los parámetros a través de la instrucción `create pfile`.

1. Crear un PFILE a partir de un SPFILE. El archivo deberá ser almacenado en la carpeta `t0204` con el nombre `e-02-spparameter-pfile.txt`. No se requiere automatizar esta tarea.

### 1.3.1.3. Obtención de los parámetros a través de v\$spparameter.

A partir de este ejercicio práctico se hará uso de la base de datos #2 creada en ejercicios anteriores. Crear un script SQL `s-01-spparameters.sql`. El script deberá realizar las siguientes acciones:

1. Entrar a sesión como `sysdba`
2. Crear un usuario `<nombre>0204` en caso de no existir. Asignar cuota ilimitada en el tablespace `users`. Asignar los privilegios correspondientes para crear sesiones, tablas, secuencias y procedimientos.
3. Crear una tabla cuyo dueño sea el usuario creado anteriormente llamada `t01_spparameters` que contenga la lista de los parámetros contenidos en `v$spparameter`. Incluir el nombre y el valor del parámetro empleando como nombres de columnas `name` y `value` respectivamente. Considerar únicamente a los parámetros que no tienen valores nulos.
4. Ejecutar el script con el usuario ordinario del sistema operativo.

```
sqlplus /nolog
startup s-01-spparameters.sql
```

Contestar las siguientes preguntas:

2. Observar que los parámetros mostrados en el archivo `e-02-spparameter-pfile.txt` tienen 2 formatos: algunos inician con `<oracle_sid>._` y otro grupo inicia con `*`. ¿Qué diferencia existe entre estos 2 grupos?
3. Comparar los 2 archivos `e-01-spparameter-alert-log.txt` y `e-02-spparameter-pfile.txt` así como el contenido de la tabla `t01_spparameters`. Confirmar que en los 3 casos, existen los mismos parámetros con los mismos valores. De encontrar diferencias mencionarlas.
4. **C3. Incluir en el reporte** las respuestas.

### 1.3.2. Modificación de los parámetros de la base de datos.

En este ejercicio se consideran los siguientes parámetros elegidos únicamente para propósitos ilustrativos. Al final del ejercicio los valores modificados serán regresados a sus valores originales.

Nombre del parámetro	Descripción
<code>cursor_invalidation</code>	Indica el estilo de validación de un cursor para sentencias DDL. DEFERRED (diferida) o IMMEDIATE (inmediata).
<code>optimizer_mode</code>	Establece el comportamiento que tendrá el optimizador para recuperar los registros de una búsqueda considerando el número de registros. FIRST_ROWS_n optimiza la recuperación de los N primeros registros. FIRST_ROWS Optimiza para obtener un conjunto pequeño de los primeros registros obtenidos. ALL_ROWS Optimiza el uso de recursos al máximo para recuperar todos los registros solicitados.
<code>sql_trace</code>	Al cambiarse a TRUE, se habilita la salida de información de debug y detalle para poder analizar y detectar problemas de desempeño en sentencias SQL.
<code>sort_area_size</code>	El tamaño máximo de memoria que se utiliza para hacer un ordenamiento. Su valor por default es 65536.
<code>hash_area_size</code>	El tamaño máximo de memoria empleado para ejecutar joins que hacen uso de tablas hash (Hash Join). Por default su valor es <code>sort_area_size * 2</code> .
<code>nls_date_format</code>	Especifica el formato de fecha por default que emplean las funciones <code>to char</code> y <code>to date</code> .
<code>db_writer_processes</code>	Especifica el número inicial de procesos que se crean para realizar las tareas del proceso de background DB Writer. Útil para sistemas que actualizan datos de forma muy frecuente.
<code>db_files</code>	Número máximo de data files que pueden ser abiertos de forma simultánea.
<code>dml_locks</code>	Indica el número máximo de DML Locks (bloqueos) que pueden existir en una transacción (un bloqueo por tabla). Su valor por default es 4 veces el valor del parámetro <code>transactions</code> .
<code>log_buffer</code>	Cantidad de memoria que se utiliza para hacer buffer de datos de REDO antes de ser enviados al REDO Log file. Cada "Redo Log Record" contiene los cambios que se aplicaron a los bloques de datos ubicados en el DB Buffer caché. Estos "Redo Log Records" son almacenados en el "log buffer" y después son escritos en el Redo Log file a través del proceso LGWR.
<code>transactions</code>	Indica el número de segmentos rollback que estarán disponibles cuando la administración de los datos undo se establece a manual (UNDO_MANAGEMENT=MANUAL).

Crear un script SQL `s-02-other-parameters.sql`. El script deberá realizar las siguientes acciones:

1. Entrar a sesión como `sysdba`.
2. Crear una tabla cuyo dueño sea el usuario creado anteriormente llamado `t02_other_parameters`. La tabla deberá contener las siguientes columnas considerando únicamente los parámetros de la lista anterior. Emplear los mismos nombres de columna excepto los que se especifican entre paréntesis: número de parámetro, nombre, valor, valor por default, bandera que indica si se puede modificar a nivel sesión (`is_session_modifiable`), bandera que indica si el parámetro puede ser modificado con la instrucción `alter system` (`is_system_modifiable`).
3. Ejecutar el script con el usuario ordinario.

```
sqlplus /nolog
startup s-02-other-parameters.sql
```

Crear un script SQL `s-03-modifica-parametros.sql` que realice las siguientes actividades:

1. Entrar a sesión como `sysdba`.
2. Aplicar los cambios solicitados a los siguientes parámetros:
  - Modificar el valor del parámetro `nls_date_format` para que las fechas se muestren en el formato `dd/mm/yy hh24:mi:ss`. Realizar este cambio únicamente a nivel de sesión.
  - Modificar el valor del parámetro `db_writer_processes` con el valor 2.
  - Modificar el valor de `log_buffer` 10 MB más del valor actual.
  - Modificar el número máximo de data files abiertos permitidos a 250
  - Modificar el número máximo de bloqueos en instrucciones DML a 2500.
  - Modificar el valor de los segmentos de `rollback` a 600.
  - Aumentar 5 MB al tamaño de la memoria para ejecutar hash joins. El cambio debe aplicarse en la sesión actual, y debe estar disponible para el siguiente reinicio.
  - Aumentar 5MB a la memoria empleada para hacer operaciones sort. El cambio solo debe hacerse a nivel de sesión.
  - Habilitar la salida de datos de debug en sentencias SQL. Realizarlo únicamente a nivel instancia.
  - Modificar la configuración para que las búsquedas de datos se realicen de la forma más óptima posible para los primeros 100 registros. Aplicar el cambio para que tome efecto de forma inmediata y sea permanente.
  - Cambiar el estilo de validación de cursores a `DEFERRED` únicamente a nivel sesión.

Recordar: 1MB = 1024 KB

Con base a los cambios en los valores de los 11 parámetros anteriores se puede observar lo siguiente:

- Algunos cambios se aplicaron a nivel de sesión. Esto se puede comprobar consultando la vista `v$parameter`
- Algunos cambios se aplicaron a nivel de toda la instancia. Esto se puede comprobar consultando la vista `v$system_parameter`
- Algunos cambios se aplicaron en el `spfile`. Esto significa que el cambio aplicado será persistido en el `SPFILE`. Esto se puede consultar en `v$spparameter`.
- En algunos ejercicios, el cambio se aplicó a varios niveles. Por ejemplo, la cláusula `scope=both`, aplica el cambio en los 3 niveles: sesión, instancia y `spfile`.

Para confirmar lo anterior, agregar las siguientes sentencias SQL al script, posterior a la modificación de los parámetros:

3. Crear una tabla llamada `t03_update_param_session` cuyo dueño sea el usuario creado anteriormente. La tabla deberá contener el nombre y el valor de los parámetros modificados a nivel sesión. Excluir valores nulos.

Para resolver este ejercicio, la sentencia SQL deberá hacer uso de la vista `v$parameter` ya que en ella se guardan los cambios a sesión:

```
--parametros modificados en la sesión del usuario
create table jorge0204.t03_update_param_session as
select name,value
from v$parameter
where name in (
  'cursor invalidation','optimizer mode',
  'sql_trace','sort_area_size','hash_area_size','nls_date_format',
  'db_writer_processes','db_files','dml_locks','log_buffer','transactions'
)
and value is not null
```

4. Considerando el ejemplo anterior, crear las tablas `t04_update_param_instance` y `t05_update_param_spfile` que guarden los valores de los parámetros modificados a nivel instancia y `spfile` respectivamente. Si las tablas ya existen, el script deberá eliminarlas y volverlas a generar.
  5. Agregar una instrucción SQL al script para crear un archivo de texto (PFILE) llamado `e-03-spparameter-pfile.txt`. El archivo deberá contener la lista de todos los parámetros configurados en el SPFILE mismos que se encuentran en la tabla `t05_update_param_spfile`. Generar el archivo en la carpeta `t0204`.
4. Ejecutar el script con el usuario ordinario.

```
sqlplus /nolog
startup s-03-modifica-parametros.sql
```

6. Ejecutar el script y realizar las siguientes acciones para confirmar resultados
  - a. Hacer una consulta a las 3 tablas creadas, y revisar que efectivamente los cambios aplicados a los parámetros son correctos. Por ejemplo, el nuevo valor del parámetro `optimizer_mode` debe aparecer en las 3 tablas; el nuevo valor del parámetro `sql_trace` no fue actualizado en `p02_update_param_spfile` ya que su valor no se modificó a nivel `spfile`. Solo aparece en las otras 2 tablas, etc. Verificar cuidadosamente los cambios ya que más adelante son verificados a través del validador del ejercicio.
  - b. Abrir el archivo `e-03-spparameter-pfile.txt` verificar que los parámetros modificados a nivel `spfile` aparezcan en él. **C4. Incluir en el reporte** el contenido del archivo únicamente para los parámetros que inicien con “\*.” (su contenido es muy corto).

#### 1.4. VALIDADOR DE CAMBIO DE PARÁMETROS.

Por la naturaleza de este ejercicio práctico, la validación se realizará en 2 partes: validación de cambios de parámetros y validación de restauración. En esta primera parte se realizará la validación de los cambios solicitados.

- Copiar todos los scripts de validación en la misma carpeta donde se ubican los scripts del ejercicio. Ejecutar el validador empleando el usuario ordinario del sistema operativo.

```
sqlplus /nolog
start s-05-oracle-validador-main-cambios.sql
```

#### 1.5. VALIDADOR DE RESTAURACIÓN DE PARÁMETROS.

Antes de iniciar el validador, realizar la siguiente actividad.

##### 1.5.1. Restauración de los parámetros de la base de datos.

Empleando alguno de los respaldos obtenidos anteriormente, realizar las acciones necesarias para restaurar los valores de los parámetros modificados en la sección anterior. Para ello crear un script SQL `s-04-restaura-parametros.sql` que realice las acciones necesarias. Tip: El script deberá detener la instancia, emplear la instrucción `create spfile` a partir del archivo `pfile` respaldado en secciones anteriores y posteriormente iniciar.

##### 1.5.2. Ejecución de validador.

- Ejecutar el validador empleando el usuario ordinario del sistema operativo.

```
sqlplus /nolog
start s-06-oracle-validador-main-restaura.sql
```

#### 1.6. CONTENIDO DE LA ENTREGA.

- C1. Código del script `s-00-crea-directorios.sh`
- C2. Código del script `e-01-spparameter-alert-log.txt`
- C3. Respuestas de las preguntas referentes al contenido de los archivos PFILE.
- C4. Contenido parcial del archivo `e-03-spparameter-pfile.txt`
- C5. Salida del validador de cambio de parámetros.
- C6. Salida del validador de restauración de parámetros.
- Entrega individual