

ADMINISTRACIÓN DE ORACLE 10g

Este manual de Administración de Oracle 10g está dirigido a todos los profesionales IT que tienen conocimientos de Oracle 10g y desean profundizar en los temas más importantes en la administración de una base de datos en Oracle 10g.

Además se han colocado ítems y tips de modo que el manual también sea de gran ayuda.

No está dirigido a las personas que están interesadas en aprender Oracle 10g desde el inicio.

ING. FRANCISCO MARTIN RICCIO CHÁVEZ, certificado en Administración de Oracle 10g.

Cualquier sugerencia o duda favor de escribir al autor al siguiente correo:

paco_jaco@yahoo.com

INDICE

| | |
|----------------------------------|----|
| ARQUITECTURA | 4 |
| CONCEPTOS | 4 |
| PARAMETROS | 7 |
| SCRIPTS | 7 |
| CREACIÓN DE BASE DE DATOS | 8 |
| CONCEPTOS | 8 |
| PARAMETROS | 10 |
| SCRIPTS | 10 |
| RECOMENDACIONES | 10 |
| IMAGENES | 10 |
| TABLESPACE - DATAFILE | 11 |
| CONCEPTOS | 11 |
| PARAMETROS | 12 |
| SCRIPTS | 12 |
| GRAFICOS | 13 |
| RECOMENDACIONES | 14 |
| SCHEMA - OBJECTS | 15 |
| CONCEPTOS | 15 |
| SCRIPTS | 15 |
| ORACLE NET SERVICES | 17 |
| CONCEPTOS | 17 |
| PARAMETROS | 18 |
| CONFIGURACIONES | 18 |
| RECOMENDACIONES | 20 |
| ERRORES COMUNES | 20 |
| IMÁGENES | 20 |
| ORACLE SHARED SERVER | 21 |
| CONCEPTOS | 21 |
| SCRIPTS | 21 |
| FORMULAS | 22 |
| RECOMENDACIONES | 23 |
| ADMINISTRACIÓN DE USUARIOS | 24 |
| CONCEPTOS | 24 |
| PARAMETROS | 25 |
| SCRIPTS | 25 |
| RECOMENDACIONES | 28 |
| AUDITORIA | 29 |
| CONCEPTOS | 29 |
| PARAMETROS | 29 |
| SCRIPTS | 29 |
| RECOMENDACIONES | 31 |
| PL/SQL | 31 |
| CONCEPTOS | 31 |
| PARAMETROS | 32 |
| SCRIPTS | 32 |
| RECOMENDACIONES | 34 |
| SQL LOADER | 35 |
| CONCEPTOS | 35 |

| | |
|-------------------------------------|----|
| PASOS..... | 35 |
| TABLAS EXTERNAS..... | 37 |
| DATA PUMP | 38 |
| CONCEPTOS..... | 38 |
| USOS..... | 38 |
| EXPORTAR | 38 |
| IMPORTAR | 39 |
| UNDO TABLESPACE - CONSISTENCIA..... | 41 |
| CONCEPTOS..... | 41 |
| SCRIPTS | 41 |
| FORMULAS | 42 |
| RECOMENDACIONES | 42 |
| BLOQUEOS..... | 43 |
| CONCEPTOS..... | 43 |
| SCRIPTS | 43 |
| TUNING..... | 44 |
| CONCEPTOS..... | 44 |
| PARAMETROS | 45 |
| SCRIPTS | 45 |
| RECOMENDACIONES | 48 |
| BACKUPS..... | 51 |
| CONCEPTOS..... | 51 |
| PARAMETROS | 53 |
| SCRIPTS | 53 |
| RMAN..... | 54 |
| IMAGENES | 54 |
| RECOMENDACIONES | 55 |
| RECOVERY..... | 56 |
| CONCEPTOS..... | 56 |
| PARAMETROS | 57 |
| SCRIPTS | 57 |
| RMAN..... | 58 |
| LOG MINER..... | 58 |
| RECOMENDACIONES | 58 |

ARQUITECTURA

CONCEPTOS

- DBA debe: Instalar y configurar el Oracle software, seleccionar el hardware, manejar la base de datos y objetos de ella, establecer backups y tuning.
- Arquitectura:
 - User Process: Esta compuesto por un user process y un server process. El user process es usado para soportar la conexión del usuario. La sesión del usuario se almacena en el Oracle Instance. Las variables e información de cada sesión se guarda en el PGA.
 - Oracle Instance: SGA + process background.
 - Redo Log Buffer: Almacena transacciones para propósitos de recovery.
 - Streams Pool: Cache de la data asociada a los requerimientos encolados
 - Background process:
 - PMON: No reubica los temporary segments cuando una sesión cerró.
 - SMON: Instance Recovery y maneja espacio en ordenamientos.
 - MMON: Recolecta estadísticas y los almacena en el AWR.
 - Job Queue (J): Ejecuta jobs de la base de datos que han sido shedule.
 - Job Queue Monitor (CJQ): Asigna jobs a la cola de jobs.
 - MMNL (Memory Monitor Light): Recolecta estadísticas para AWR.
 - MMAN (Memory Manager): Es activado en el Database Control y esto permite que el ADDM ajuste database buffer cache según los estadísticas. El MMAN a cada componente del SGA lo dimensiona.
 - Recovery Writer (RVWR): Escribe recovery information cuando se habilita el flash recovery.
 - Change Tracking Writer (CTWR): Mantiene tracks de los bloques cambiados desde un backup incremental por el RMAN.
 - DBWriter baja a disco en las siguientes circunstancias:
 - No hay espacio libre en el buffer cache para nueva data.
 - Demasiados commits que no han bajado a disco.
 - Usando shutdown o usando checkpoint.
 - El tablespace es puesto en modo backup, offline o readonly.
 - Un segmento es borrado.
 - LOGWriter baja a disco en las siguientes circunstancias:
 - Cada 3 segundos.
 - Cuando se hace un commit.
 - Cuando el redo buffer tiene 1 MB de redo información.
 - Checkpont.
 - LRU es el algoritmo para manejar el contenido del Shared Pool y del Database Buffer Cache.

- Cada componente del SGA (Buffer Cache, Shared Pool, etc) trabajan bajo granulas, donde una granula equivale a:
 - 4 mb, si el SGA_MAX_SIZE < 128.
 - 16 mb, si el SGA_MAX_SIZE >= 128.
 Mínimo el SGA necesita 12 MB, es decir una granula para el Buffer Cache, 1 granula para el Shared Pool y una granula para el Fixed Cache.
- PFILE es llamado init(nombre de la instancia).ora y spfile(nombre de la instancia).ora tiene mas de 250 parámetros configurables.
- NLS_LANGUAGE: Especifica el default del idioma.
- NLS_TERRITORY: Especifica el default del territorio.
- OPEN_CURSORS: Especifica el máximo números de cursores de una sesión puede abrir al mismo tiempo.
- SESSIONS: Especifica el máximo número de sesiones al conectarse SGA_TARGET.
- Base de datos:
 - Password file: Usado para saber quienes están garantizados con los privilegios de SYSDBA y SYSOPER.
 - System tablespace: Almacena las tablas del diccionario de datos.
 - SYSAUX: Almacena segmentos para el AWR, Online Analytical Processing (OLAP).
 - Temp: Operaciones de ordenamiento, es requerido cuando el tablespace es almacenado como local managed sino es opcional.
- Oracle Application 11i es una especie de SAP.
- Oracle Collaboration Suite integra todas las comunicaciones, mail, faxes, wireless, conectividad y conferencias web.
- Oracle soporta 1000 columnas y billones de registros en una tabla.
- Oracle tiene sus system tables que almacenan la información de las tablas del usuario, tipo de datos, permisos, etc. A esas tablas se les conoce como metadata y sus nombres están encriptados como obj\$ y sus columnas también como dataobj#;
- Tipos de vista de la metadata:
 - Data dictionary: Puede ser más de 1300 vistas. Comienzan con dba, all, user. Ejemplo: dba_users, dba_views, dba_tab_columns, dba_constraints, dba_cons_columns, etc.
 - Vistas dinámicas: Son más de 350 vistas dinámicas. Inician con v\$, ejemplo: v\$version ó v\$option.
 - Diferencia entre ellos: El data dictionary solo se acceden cuando la base de datos esta en estado open. El data dictionary se mantiene si se apaga la base de datos. El data dictionary devuelve en mayúscula sus resultados.
- Las vistas del diccionario de datos fueron creado por el script catalog.sql. Ojo: Las tablas del diccionario de datos almacenan la definición de todos los objetos como tablespaces, vistas, datafiles, tablas, programas y su tamaño de espacio. También las clausulas de integridad, usuarios, roles, permisos y auditoria.
- El unique permite valores nulos pero el primary key no.
- DELETE CASCADE al crear la tabla permite que cuando sea borrado el padre borre al hijo.
- Un segmento es una entrada que consume espacio físico. Ejemplo: Tablas, índices, rollback o undo. Un segmento puede tener hasta 2 billones de extendidos.

- Lo mínimo de un extendido es 5 database block.
- Los database block pueden ser de: 2 kb, 4kb, 8 kb, 16 kb y 32 kb, etc.
- Un bloque de sistema operativo puede ser de 512 bytes a 2 kb dependiendo del sistema operativo.
- Oracle Discover es un ayudante para consultas via web y Oracle Forms & Reports permite acceder a la base de datos usando formularios web y reportes.
- PL/SQL permite testing, loops y excepciones.
- OCI (Oracle Call Interface) permite a los lenguajes de programación interactuar con la base de datos.
- Instalación:
 - Mínimo debe tener la pc con 512 mb de ram, 1 gb de swap, /temp de 300 mb, 1.5 gb es requerido para la instalación. 1 gb más es requerido para crear una base de datos con el DBCA.
 - Al modificar el kernel de Linux se realiza lo siguiente:
 - Especificar el número de procesos que puedan correr en el servidor.
 - Máximo numero de archivos abiertos en la base de datos.
 - Máximo tamaño permitido para segmentos de memoria.
 - OFA:
 - Nombramiento de archivos de Linux y puntos de montaje (String + numbers).
 - Nombramiento de directorios.
 - Nombramiento de archivos de base de datos.
 - Localizaciones de los archivos.
 - El grupo Oracle de Linux son llamados dba.
 - Carpetas creadas para administración para una base de datos:
 - adump: trace de auditoria.
 - bdump: alert + process background.
 - cdump: Los procesos del sistema operativo escriben sus trace.
 - udump: Traces de los usuarios.
 - create: Tiene el script de la creación de la base de datos.
 - Variables del sistema operativo:
 - \$TWO_TASK ó %LOCAL%: Establece el default conexión.
 - \$PATH: Dice donde están los binarios de Oracle.
 - LD_LIBRARY_PATH: Apunta a \$ORACLE_HOME/LIB y apunta a la localización de las librerías de objetos compartidos.
 - runinstaller –ignore SysPrereqs: No valida las configuraciones adecuadas en el Linux.
 - OraInventory: Mantiene un track de los productos instalados de Oracle en el servidor.
 - sh orainstRoot.sh (corriendolo como root): Crea directorios que son usados para soportar la instalación de Oracle y entrega los permisos adecuados a ellos. Tambien crea un archivo llamado orainst.loc el cual indica al Oracle Installer la localización del inventory localtion.
 - sh root.sh (corriendolo como root): Copia algunos archivos afuera y dentro del \$ORACLE_HOME y entrega los permisos adecuados de ellos. Crea el archivo ORATAB.
 - Oratab file es creado en /etc ó /var/opt/oracle cuando se instala Oracle. Es utilizado para listar las bases de datos y las versiones de software instaladas en el servidor.
Tiene las siguientes entradas: database_sid:oracle_home_dir:Y|N
El flag indica si debe iniciarse la base de datos junto en el arranque del servidor.

PARAMETROS

| Parámetro | Objetivo |
|-----------------|---|
| MAX_SGA_SIZE | Especifica el tamaño del SGA. |
| CORE_DUMP_DEST | Localización del CDUMP. |
| AUDIT_FILE_DEST | Donde van los trace de auditoria. |
| DBA_TEMP_FILES | Muestra los tablespaces temporary. |
| SGA_TARGET | Si es <> 0 el toma el Oracle toma el tamaño de SGA y él redistribuye a cada parte del SGA como él lo crea mejor. Si colocamos algún valor en los componentes del SGA, indica que al menos tendrá ese valor el componente. |

SCRIPTS

| Objetivo | Script |
|---|--|
| Obtiene el spfile del pfile. Obtiene el pfile del spfile. | create pfile from spfile; create spfile from pfile; |
| Muestra el tamaño del SGA. | v\$sga. |
| Muestra cada componente del SGA. | v\$sga_dynamic_components. |
| Muestra las configuraciones del pfile ó spfile. | v\$parameter ó v\$sppparameter. |
| Muestra las propiedades de la base de datos. | database_properties. |
| Tamaño del buffer cache. | v\$buffer_pool. |
| Altera el tamaño del SGA. Además cuando es diferente a 0 habilita a Oracle para que administre cada componente automáticamente. | alter system set sga_target = #; |

CREACIÓN DE BASE DE DATOS

CONCEPTOS

- Oracle Enterprise Management Framework provee un conjunto de herramientas para tareas típicas de administración.
- El Agent es el responsable de comunicar un servidor con el Oracle Enterprise Management. El Agent es el responsable de coleccionar información de un servidor, el agent se corre sobre cada servidor que se desea monitorear vía Oracle Enterprise Management Framework. El Agent almacena la información recolectada en el Management Repository. El Agent se comunica mediante el Management Service.
- Managed Targets son los servidores que uno desea administrar.
- Management Service es un componente web basado en Java que es la interfaz para monitorear y controlar los Managed Targets con el Management Framework.
- Management Repository monitorea la información recolectada acerca de un Managed Target que es almacenada en el Management Repository. Comprende 2 tablespaces que contiene información sobre los administradores, targets y aplicaciones que son administradas.
- Oracle Enterprise Manager 10g Grid Control es una interfaz web que comunica con todos los componentes de la empresa. Se administra hosts, databases, listeners, application servers, etc.
- Database Control permite administrar y monitorear una instancia Oracle o un Application Real Cluster.
- Application Server Control permite monitorear una instancia de un Oracle Application Server.
- Las base de datos pueden ser administrada centralizadamente mediante un Grid Control (Ahí debemos levantar el Agent por cada servidor) o se puede administrar invididualmente por el Database Control.
- Las plantillas esta guardadas en formato XML. La plantilla muestra información como los parámetros de inicialización, character sets, control files, tablespaces, datafiles y redologs.
- El DBCA permite hacer notificaciones al crear una base de datos notificando cuando un thresholds son alcanzados vía email, también permite generar backups diarios (solicita un usuario del sistema operativo y su password).
- Los usuarios por defecto son: SYS (Es el owner de todas las tablas internas del Oracle que constituyen el data dictionary), SYSTEM (Permite tareas administrativas), DBSNMP (Es el usuario usado para monitorear y recolectar estadísticas de performance en la base de datos) y SYSMAN (Es equivalente al usuario SYS pero en el Database Control).
- Los mecanismos de almacenamiento que presenta DBCA para almacenar los archivos de la base de datos son: File System Storage (El sistema operativo se encarga del manejo de los archivos, el DBCA utiliza OFA para la estructura de la base de datos), ASM (Agrupa discos en grupos lógicos como unidades de almacenamiento y permite mirroring y stripping) y RAW DEVICES (Oracle es el encargado de escribir y leer nativamente el disco de almacenamiento).
- En el DBCA debemos especificar el lugar de nuestro flash_recovery_area y podemos también habilitar la opción de archiver log.
- Con el DBCA podemos setear el tamaño del SGA y del PGA, especificar el tamaño del db_block_size, el máximo número de procesos simultáneos del sistema operativo que pueden ser conectados al servidor Oracle (Se debe ingresar 6 procesos como mínimo por cada proceso background del Oracle), el

- character set (toma por default la configuración del sistema operativo) y el tipo de conexión (dedicated o shared).
- Las plantillas se almacenan en \$ORACLE_HOME/assistants/dbca/templates. Las plantillas pueden ser de 2 tipos: SEED (Contiene definición de la base de datos y datafiles con redologs, tiene la ventaja es que el DBCA copia los datafiles y redolog en el archivo de definición y estos son prerreconstruidos haciendo mas rápida la creación. Su extensión es dbc y los archivos asociados a los redo logs y datafiles son de extensión djf y esto solamente hace que el usuario modifique el nombre de la base de datos, la localización de los datafiles, redo logs, control files y parámetros de inicialización), NOSEED (Contiene definición personalizada, esto no viene con datafiles y redo logs pre configurados, su extensión es bdt).
 - Los archivos djf contienen los redologs y datafiles predefinidos en un seed template.
 - Con el DBCA se puede crear plantillas en base a una plantilla preexistente, de una base de datos (solamente estructura) y de una base de datos con estructura y datos.
 - El log de los Database Control se almacena en: \$ORACLE_HOME/(hostname)/sysman/log.
 - La lista de puertos utilizados se encuentra en: \$ORACLE_HOME/install/portlist.ini.
 - El pfile y spfile se ubica en \$ORACLE_HOME/dbs/.
 - El estado NOMOUNT comienza la instancia sin levantar la base de datos. El spfile o pfile es leído y los procesos background son iniciados. Sirve para crear una base de datos.
 - En el estado MOUNT es leído el control file y sirve para recovery, mover files y poner la base de datos en archive log.
 - El estado STARTUP FORCE se usa si estamos en dificultades en comenzar la base de datos, si la base de datos esta abierta le hace shutdown abort (bajada sucia) y levanta la base de datos.
 - El estado STARTUP RESTRICT está en modo OPEN y entrega solamente acceso a los usuarios que tiene el privilegio RESTRICTED SESSION.
 - SHUTDOWN NORMAL, nuevas conexiones no son permitidas y la base de datos espera que todos los usuarios se desconecten.
 - SHUTDOWN TRANSACTIONAL, nuevas transacciones no son permitidas ni nuevas conexiones, una vez que todas las transacciones son completadas las conexiones son desconectadas.
 - SHUTDOWN IMMEDIATE, las transacciones son rollback y las sesiones son desconectadas.
 - SHUTDOWN ABORT, uncommit data no se rollback.
 - El alert log registra los siguientes sucesos:
 - Startup y Shutdown de la base de datos.
 - Ciertos tipos de tareas administrativas hechas con alter system y alter database.
 - Cierta tipo de errores de base de datos (ORA).
 - Los valores de inicialización que son diferentes a su default.
 - Si el SYSTEM tablespace esta siendo usado como TEMPORARY tablespaces. (Esto sucede cuando no hay un TEMPORARY tablespaces asignado).
 - El archivo oratab ubicado en /etc/ es copiado en el proceso de instalación por el root.sh. En este archivo se ubica el servicio de bd (\$ORACLE_SID), el \$ORACLE_HOME y un flag que indica si la base de datos debe iniciarse con el sistema operativo.

- Cuando borramos una base de datos con el DBCA esto borra también los files físicamente.
- Cuando creamos una base de datos en la instalación y seleccionamos OMF (Oracle Enterprise Manager) permite que el Database Control y Database Grid Control estén disponible.
- Para instalar el Database Control de una base de datos que no ha sido especificado con OMF se debe realizar una serie de pasos manualmente.
- Para configurar una base de datos para ASM debemos aplicar el comando:
emca -a.

PARAMETROS

| Parámetro | Objetivo |
|-------------------------------|---|
| BACKGROUND_DUMP_DEST = 'ruta' | Especifica la ruta del alert log (BDUMP). |

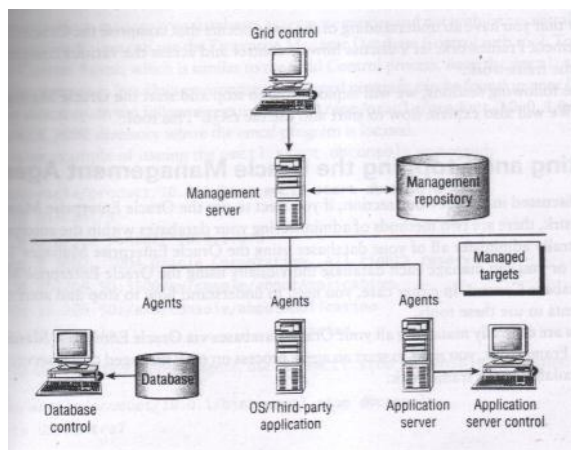
SCRIPTS

| Objetivo | Script |
|--|--|
| Para ver el status del Database Control o del Agent. | emctl status (agent ó dbconsole). |
| Borrar una base de datos. Elimina todos los archivos asociados si no usa RAW DEVICES. Quizas no elimina los archived logs. | sqlplus sys/password as sysdba ó sysoper startup mount; drop database; |
| Deshabilita la opción de restrict a la base de datos. | alter system disable restricted session; |
| Levanta otra configuración en base a un spfile o pfile. | startup modo (nomount, mount ó open) pfile ó spfile = 'ruta'; |
| Sube la instancia a un estado. | alter database mount ó open; |

RECOMENDACIONES

- Se debería tener el flash_recovery_area en otro disco para tener mayor seguridad y performance.
- Para aplicaciones transaccionales se debería setear el db_block_size a 8 kb y para data warehouse a 16 kb a más.
- Limpiar periódicamente el archivo alert_log puede llenar disco duro.

IMAGENES



TABLESPACE - DATAFILE

CONCEPTOS

- Un tablespace contiene 0 o mas segmentos (Cada segmento reside en un tablespace).
- Un segmento son objetos de esquema, los tipos son: tablas, índices, temporales y rollback; ellos son almacenados fuera del data dictionary. Los constraint y sequences son almacenados en el data dictionary por lo tanto no son segmentos.
- Cada segmento contiene extents. (Cada extents reside en un datafile).
- Una base de datos puede tener maximo 64000 datafiles.
- Un extent contiene data blocks. (Cada data block es un conjunto de bytes de hd).
- SYSTEM y SYSAUX tienen el tamaño default de data block.
- Cada tablespace puede tener diferentes tamaños de data block.
- El tablespace SYSTEM almacena el data dictionary.
- El tablespace SYSAUX almacena objetos para el catalogo del RMAN, el AWR, Data Mining y otros. Fue pensado para reducir objetos que eran colocados en el SYSTEM. Se puede aumentar su tamaño en cualquier momento.
- Opciones de tablespace: Big file o small file, manage extents locally o dictionary y manage segment space automatico o manual.
- Los tablespace bigfile tienen 1 datafile, su tamaño maximo es (4 TB * Tamaño de un data block, Ejemplo Si 8 KB es el tamaño del data block de un tablespace su tamaño máximo seria 32 TB).
- Los tablespace smallfile pueden tener de 1 a 1022 datafiles para un tablespace. Su tamaño maximo es (4 GB * Tamaño de un data block, Ejemplo Si 8 KB es el tamaño del data block de un tablespace su tamaño máximo seria 32 GB). SYSTEM y SYSAUX siempre son smallfiles tablespaces.
- OMF (Oracle Managed Files), permite que Oracle se encargue de nombrar y ubicar a nuestros datafiles automáticamente. Por defecto asigna 100 MB y autoextendido.
- Los tablespace pueden ser Data Dictionary (Utiliza un registro de los extents libres y usados mediante las tablas FET\$ y UET\$ en modo recursivo, los extendidos son manejados vía data dictionary provocando contención en el SYSTEM tablespace), Local (Default y utiliza bitmaps (conjunto de bloques) para manejar los extendidos para dejar de utilizar el modo recursivo). Local tiene 2 tipos: UNIFORM (Mantiene los extents del mismo tamaño, default 1 MB, no puede ser aplicado a los undo tablespace y es el default para los Temporary Tablespace) y AUTOALLOCATE (default). Se puede convertir de de Data Dictionary a Local y viceversa. Pero no se puede convertir el Local a Data Dictionary el tablespace SYSTEM y los Temporary Tablespaces.
- Un tablespace Local puede ser MANUAL (Utiliza PCT_FREE (insert DML) y PCT_USED (delete o update DML), es el default y es fijo en los Temporary Tablespace y en el SYSTEM Tablespace) o AUTOMATIC (Utiliza bitmaps).
- Los Temporary Tablespaces segments son generados por un order by, group by, create index e inserts a tablas temporales. Los temporary tablespaces son seteados a NOLOGGING. No se puede renombrar un temp file o ponerlo read only. Lo que es válido es llevar el temp file a online u offline pero no el tablespace.
- Undo tablespace sirve para rollback, reconstruir un read-consistent y recover de corrupciones lógicas.
- Un tablespace debe ser llevado a offline para un recover de el tablespace o mover datafiles.

- SYSTEM, UNDO y SYSAUX no pueden ser colocados readonly.
- Un tablespace al ponerlo en offline tenemos 3 opciones: NORMAL, TEMPORARY e IMMEDIATE. El NORMAL hace un checkpoint y asegura que sea escrito en disco, es ideal en NOARCHIVELOG y al ponerlo online no requiere media recovery. El TEMPORARY también hace checkpoint pero no asegura que se haya hecho a todos los datafiles por lo tanto puede ser que haya un media recovery en alguno de los datafiles. IMMEDIATE no hace checkpoint y ejecuta un media recovery al volver a colocar el tablespace en online. El default es normal.
- En versiones inferiores de Oracle 10g R2 no se puede eliminar un datafile.
- El comando ALTER DATABASE DATAFILE 'ruta' OFFLINE DROP indica que el datafile se pondrá en estado offline con la intención de eliminar el tablespace.
- No se puede renombrar el SYSAUX y SYSTEM tablespace.
- Cuando no hay un tablespace temporal en la base de datos, Oracle utiliza el SYSTEM tablespace en reemplazo de este.

PARAMETROS

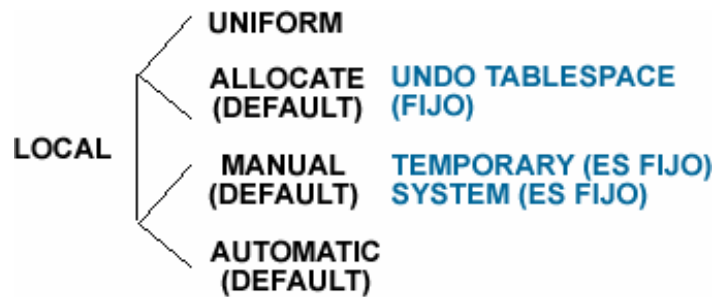
| Parametro | Objetivo |
|----------------------------|---|
| db_block_size = valor | Permite especificar el tamaño default de un data block. |
| db_create_file_dest = ruta | Habilita OMF y especifica el lugar donde Oracle creara los datafiles. |
| db_create_online_dest_n | Donde n = 1..5 y es donde se ubicarán los redo logs y control files utilizando OMF. |
| undo_management = nombre | Define el undo tablespace de la base de datos. |
| sort_area_retained_size | Tamaño ocupado por ordenamientos. |

SCRIPTS

| Objetivo | Script |
|---|--|
| Crea un big tablespace. (Quitando la palabra bigfile se convierte en un smallfile). Lo puede crear Local o Uniform. Opcionalmente, al datafile se le puede habilitar la opción de autoextend. | create <i>bigfile</i> tablespace nombre datafile 'ruta1' size #, 'rutan' size # <i>extent management local autoallocate ó uniform segment space management auto ó manual</i> |
| Crea un tablespace en OMF. | create tablespace nombre ó create tablespace nombre datafile size #; |
| Crea un Temporary Tablespace. | create temporary tablespace nombre tempfile 'ruta' size # <i>autoextend on next #(K, M, G) maxsize unlimited;</i> |
| Crea un Undo Tablespace. | create undo tablespace nombre datafile 'ruta' size #; |
| Seteando el default Temporary Tablespace. | alter database default temporary tablespace nombre; |
| Redimensiona un datafile. | alter database datafile 'ruta' resize #; |
| Habilita el autoextend en un datafile. Puede ingresarse una cantidad máxima o usar la constante UNLIMITED. | alter database datafile 'ruta' autoextend on next # maxsize # ó autoextend on maxsize #; |
| Elimina un tablespace. Elimina un tablespace con sus segmentos (tablas, índices, etc). Elimina un tablespace con sus segmentos | drop tablespace nombre; drop tablespace nombre including contents; drop tablespace nombre including contents and datafiles; |

| | |
|---|---|
| y datafiles físicamente. Para eliminar un tablespace sus objetos n deben estar en uso. | |
| Renombrar un tablespace. | alter tablespace nombre1 rename to nombre2; |
| Agregar datafiles a un tablespace. | alter tablespace nombre add datafile 'ruta' size #; |
| Reubicar un datafile. (Es el mejor modo de renombrar un datafile porque no ponemos el tablespace completamente en offline). Solo se puede hacer en modo archiver. | alter database datafile 'ruta1' offline; Mover el datafile con el sistema operativo. alter database rename file 'ruta1' to 'ruta2'; recover datafile 'ruta2'; (Sincroniza la cabecera del datafile con la base de datos). alter database datafile 'ruta2' online; |
| Eliminando un datafile. (Valido en 10g R2) | alter tablespace nombre drop datafile 'ruta'; |
| Pone offline u online un tablespace y renombrando sus datafiles. Solo se puede hacer en modo archiver la opción de renombrar un datafile. | alter tablespace nombre offline; Mover un datafile con el S.O (u otra tarea). alter tablespace nombre rename datafile 'ruta1' to 'ruta2'; alter tablespace nombre online; |
| Pone read only a un tablespace. | alter tablespace nombre read only; ó alter tablespace nombre read write; |
| Vistas. | dba_tablespaces (buena información), dba_data_files y dba_temp_files. |
| Muestra información de los Temporary Tablespace. | dba_temp_files ó v\$tempfile. |
| Añade un Temp File al Temporary Tablespace. Elimina un Temp File del Temporary Tablespace. Pone un Temp File en OFFLINE u ONLINE. | alter tablespace nombre add tempfile 'ruta' size #; alter database tempfile 'ruta' drop; alter database tempfile 'ruta' offline ó online; |
| Convierte de Data Dictionary a Local. | execute dbms_space_admin.tablespace_migrate_to_local(t ablespace_name =>'nombre'); |
| Convierte de Local a Data Dictionary. | execute dbms_space_admin.tablespace_migrate_from_loc al(ablespace_name =>'nombre'); |
| Permite visualizar los datafiles asociados a su tablespace. | dba_data_files; |
| Obtiene el PL/SQL de un tablespace. | SELECT DBMS_METADATA.GET_DDL('TABLESPACE','no mbre') FROM DUAL |
| Obtiene el esquema de una tabla en formato XML. | SET LONG # SELECT DBMS_METADATA.GET_XML('TABLE','nombre') |

GRAFICOS



RECOMENDACIONES

- Tener los índices en diferentes tablespace que las tablas y cada tablespace (sus datafiles) en diferente discos duros.
- A veces es recomendable usar tablespace bigfile porque reduce el tiempo de actualización de cabeceras en un checkpoint al existir menos datafiles.
- Para crear un Temporary Tablespace es mejor primero crear un tablespace normal, luego eliminarlo (drop tablespace nombre) y luego crear el Temporary Tablespace pero usando reuse, ejemplo: create temporary tablespace nombre tempfile 'ruta' size # reuse. Con la finalidad que se guarde un espacio desde que se crea el tempfile ya que un Temporary tablespace por default se reserva la primera vez que se hace una operación de ordenamiento y no cuando se crea, haciendolo la primera vez un proceso lento.
- Si nosotros deseamos borrar un datafile que fue agregado por equivocación a un tablespace y este datafile aun no se le ha agregado objetos, la mejor manera de eliminarlo es redimensionar ese datafile a un tamaño menor a 5 bloques de Oracle. Con esto hacemos que nunca sea utilizado para almacenar objetos y luego podemos excluirlo de la base de datos con el tiempo.
- Para hacer un backup de un tablespace:
alter tablespace nombre begin backup
copiar los datafiles.
alter tablespace nombre end backup
Al tenerlo en begin backup permite que se siga copiando data en los datafiles pero se actualizará las cabeceras del datafile.

SCHEMA - OBJECTS

CONCEPTOS

- El tipo de dato LONG no permite que la tabla sea particionada y no puede ser utilizado en subqueries.
- CLOB es el tipo de dato alfanumerico mas grande que el VARCHAR2 y reemplaza al tipo de dato LONG.
- BLOB almacena de forma binaria (muy recomendable) y BFILE es un puntero a un archivo.
- El tipo de dato RAW permite registrar datos binarios de hasta 2000 bytes especificado entre paréntesis, el LONGRAW almacena datos de longitud variable hasta una capacidad de 2 GB.
- Un numeric(x,y) indica que x-y dígitos es la parte entera que puede almacenar e y dígitos en la parte decimal.
- ROWID es un tipo de dato que almacena un valor hexadecimal que representa la dirección única de una fila en su tabla.
- Las tablas, vistas, secuencias, synonyms, procedures, functions, packages and data types estan en el mismo namespace.
- No se puede aplicar unique constraint, primary key o foreign key sobre campos CLOB, LONG; BLOB, RAW and timestamp with timezone.
- Cuando se utiliza un nologging en una tabla o índice esto permite que ciertas transacciones no generen redo log tales como: alter index rebuild, create table as select y particiones principalmente. Toda operación DML generará redo a pesar que la tabla este en nologging. Si un tablespace tiene nologging por defecto sus objetos lo serán a menos que el objeto se especifique que será lo contrario, ejemplo: create table tabla1 nologging as select * from tabla2.
- En el EM Database Control se puede administrar las tablas pero no se puede generar los DML que han sido comiteados en las tablas ni tampoco se puede particionar las tablas desde la consola gráfica.
- **Cuando se reubica una tabla a otro tablespace logra que sus estadísticas sean inválidas de modo que consigue una bajada de performance.**

SCRIPTS

| Objetivo | Script |
|--|--|
| Crea una tabla en un tablespace determinado. El nologging compress permite que ciertas transacciones no vayan a los redolog y lo que almanace en la tabla sea comprimido. | create table nombre_tabla (...) nologging compress tablespace nombre_tablespace |
| Permite crear tablas temporales. Ambas perduran pero la información perdura antes de un commit ó al cerrar la sesión respectivamente. | create global temporary table nombre_tabla (...) on commit delete rows ó on commit preserve rows; |
| Permite asignar un comentario a una tabla o a una columna. | comments on <i>table</i> ó <i>column</i> is 'texto'; |
| Permite visualizar los comentarios de las tablas y columnas. | dba_tab_comments ó dba_col_comments; |
| Renombra una tabla. | alter table nombre1 rename to nombre2; |
| Envia una tabla a otro tablespace. | alter table nombre_tabla move tablespace nombre_tablespace; Luego reconstruir sus índices. |
| Permite borrar el detalle de una tabla o lo | alter table nombre_tabla add constraint (...) |

| | |
|--|--|
| pone en null a ese foreign key. | on delete cascade ó on delete set null; |
| Crea un índice en un tablespace determinado. Crea un índice de tipo bitmap (Poco valores ejemplo, sexo: m-f. Usado para datawarehouse), existe btree tambien. | create index nombre_indice (...) tablespace nombre_tablespace create bitmap index (...) |
| Reconstruye un índice y encoje su tamaño (lo hace más eficiente). Nota: no utilizar coalesce en vez de rebuild, esto no lo hace al 100% la reconstrucción. Move el índice a otro tablespace. | alter index nombre_indice rebuild; alter index nombre_indice rebuild tablespace nombre_tablespace; |
| Renombra un tablespace. | alter index nombre1 rename to nombre2; |
| Recrea una vista. | alter view compile; |
| Crea una secuencia. | create sequence nombre increment by # start with # # ó NOMAXVALUE (10^{27}) # ó NOMINVALUE (-10^{26}); |
| Visualizar las columnas de una tabla. | desc tabla ó dba_tab_columns. |
| Habilitando los constraint. En el caso de disable: <ul style="list-style-type: none"> • Deshabilita los constraint. • Elimina los índices. • Permite valores que violen el constraint. | alter table table primary key disable ó enable validate; |

ORACLE NET SERVICES

CONCEPTOS

- Oracle NET es el responsable de comunicar el cliente y el servidor y viceversa.
- Puede estar configurado en el cliente.
- Oracle NET soporta productos middleware como: Oracle Application Manager y el Oracle Connection Manager.
- Oracle NET soporta una variedad de protocolos como TCP y named pipes.
- LDAP es un protocolo que permite el acceso a un servicio de directorio ordenado y distribuido para buscar información en un entorno de red. Habitualmente almacena información de login y password e información personal del usuario. Permite gestionar de forma centralizada la seguridad de una red. Algunas implementaciones de esto es el Active Directory, Novell Directory, RedHat Directory Server y Oracle Internet Directory (<http://es.wikipedia.org/wiki/LDAP>)
- Oracle Internet Directory utiliza Directory Naming para resolver naming centralizados. Ojo: el cliente solicita la cadena de conexión al Oracle Internet Directory y este le envía la cadena y con la cadena el cliente puede conectarse al servidor Oracle.
- Oracle Connection Manager es una solución middleware (Se ubica entre los clientes y el servidor) y provee ciertas características como: Multiplexación (Agrupa conexiones de cliente y los envía a ellos como una conexión multiplexada), Acceso a la red (Permite establecer reglas para restringir IP's) y además permite usar diferentes protocolos al comunicarse con el cliente y actuar como traductor al servidor. En él corren 2 procesos background llamados CMGW y CMADMIN process.
- Oracle Advanced Security provee herramientas de seguridad para transmitir la información (utiliza mecanismos de encriptación) y mecanismos de autenticación confiable a los usuarios en el Oracle Enterprise.
- Firewalls: Existen 2 categorías: IP-Filtering (Filtra los paquetes que no vienen de un grupo de máquinas) y Proxy-Based (Filtra los paquetes cuyas cabeceras no corresponden a una regla, no es por máquina).
- EXTPROC process es una llamada del Oracle a un procedimiento externo escrito en un lenguaje de programación. El listener es el responsable de abastecer información (nombre del procedimiento, parámetros, etc) al procedimiento.
- Un listener puede escuchar para diversas bases de datos y diversos protocolos.
- Los listeners deben tener diferentes nombres y puertos entre ellos.
- El registro de servicio dinámico ocurre cuando una base de datos utiliza un listener con la finalidad que los usuarios se puedan conectar a este, sin que estuviera especificado en el listener.ora. Cuando se especifica en el listener.ora se llama registro estático (Default en 8i). PMON es el responsable de registrar la información en el listener, cuando es dinámico. No se verá la asociación del listener a la base de datos en el listener.ora, solamente esto podrá ser visto desde un lsnrctl services.
- Los modos que responde el listener son dedicated connection (direct y redirect (cuando el cliente se conecta desde una diferente pc remota) y shared connection (direct y redirect). Cuando es vía dedicated connection el usuario resuelve el nombre y el listener comienza un dedicated process y envía al cliente la dirección del dedicated process y el cliente establece la comunicación. Lo mismo sucede en un shared connection la diferencia que el listener envía la dirección al cliente de un dispatcher, el cliente establece la comunicación con el dispatcher y este confirma y luego el PMON actualiza al

listener indicándole la cantidad de conexiones esta siendo servidas por el dispatcher.

- El log del listener almacena las subidas y bajadas del listener, las conexiones que fueron y no fueron satisfactorias y los tipos errores de conexión. Los campos del log están separados por un asterisco.
- Cada paquete en el servidor cuando se habilita el trace tiene un tipo, los cuales pueden ser: 2 (accept), 4 (refuse), 5 (redirect), 7 (data vacía), 9 (abort), 11 (resend) y otros.
- Oracle ofrece 5 métodos para la resolución del nombre del servicio: Oracle Internet Directory (Directory name), External Naming (Utiliza un nombre de resolución externo tal como NIS - Network Information Service, el cual almacena la información del servicio de la base de datos), Hostnaming, Oracle Easy Connect y Localnaming.
- Hostnaming, son para pequeñas redes, reduce la cantidad de configuración necesaria, permite olvidarse de la configuración del cliente.
Requisitos: Usar TCP/IP, no se debe usar características de red avanzada como Oracle Connection Manager, debe existir una resolución de nombres tal como un DNS o hosts files además **lo más importante que el listener debe estar seteado con el parametro GLOBAL_DBNAME con el nombre de la maquina (Se configura en la sección SID_LIST_NOMBRE_LISTENER).** El cliente no requiere configuración, solo ingresar en el sqlnet.ora (HOSTNAME). Si se tiene varias base de datos en una misma maquina hay que especificar en el archivo hosts del cliente 2 entradas con el mismo ip pero diferente alias además cada listener.ora ingresar en el GLOBAL_DBNAME el nombre del alias correspondiente y el servidor debe tener varios nombres de resolución.
Para conectarse: user/password@hostservidor.
- En Easy Connect si el nombre de la instancia es igual al nombre del servidor se omite el nombre del servicio al igual si esta usando el puerto 1521 por default.
- El default localización de los archivos de configuración del cliente se puede cambiar de la variable de entorno TNS_ADMIN (En Windows: HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE).
- Localnaming permite varios protocolos como TCP/IP, Named Pipes, IPC, etc.
- El tnsping nos muestra el tiempo que se demora un cliente para comunicarse con el listener de ida y vuelta (round-trip).
- Un servicio denominado unqualified net service es un servicio que no provee un nombre de dominio.
- Cuando el listener no esta levantado no se puede entrar al Database Control. Muestra el siguiente error (The network adapter could not establish the connection).

PARAMETROS

| Parametro | Objetivo |
|----------------|---|
| local_listener | Permite especificar la localización del listener que lo registrará. |
| instance_name | Indica el nombre del servicio de la base de datos. |
| service_names | Indica el nombre del servicio acompañado del dominio. |

CONFIGURACIONES

| Objetivo | Script |
|----------------------|--|
| Partes del listener. | <ul style="list-style-type: none"> • NOMBRE_LISTENER = Va el protocolo, host y el puerto. |

| | |
|--|--|
| | <ul style="list-style-type: none"> • SID_LIST_NOMBRE_LISTENER = Va a que servicio de base de datos debe enviar, el global db_name y el oracle_home. (Registro estático). |
| Registro dinámico de servicios. | <p>El parámetro LOCAL_LISTENER setearlo como: set LOCAL_LISTENER = "(ADDRESS_LIST = (Address = (Protocol = <i>protocolo</i>) (Host = <i>pc</i>) (Port = #)))</p> <p>Configurar los parámetros INSTANCE_NAME (nombre del servicio de la base de datos) y SERVICES_NAMES (nombre del servicio más el dominio) si no lo estuvieran.</p> |
| Parámetros opcionales del listener. | <ul style="list-style-type: none"> • INBOUND_CONNECT_TIMEOUT_NOMBRE_LISTENER (Especifica cuanto tiene que esperar un listener para validar el inicio de una sesión, el default es 10 segundos). • LOG_FILE_NOMBRE_LISTENER (Especifica donde se escribira el log del listener, por default se guarda en \$ORACLE_HOME/network/log/listener.log). • TRACE_LEVEL_NOMBRE_LISTENER (Especifica el nivel del trace de las conexiones, valores: off, user, support y admin). • TRACE_FILE_NOMBRE_LISTENER (Especifica al archivo donde se registra los traces del listener). |
| Comandos sobre el listener. | <p>lsnrctl status.- Muestra si el listener esta activo y la localización de los log y trace files.</p> <p>lsnrctl services.- Muestra información acerca de los servicios.</p> <p>lsnrctl help.- Muestra todos los comandos disponibles.</p> |
| Opciones sobre el listener. | <p>lsnrctl> save_config (Copia la configuración al listener).</p> <p>lsnrctl> trace 0 ó 1 ó 2. (Habilita el trace).</p> <p>lsnrctl> show (Muestra todas las opciones disponibles).</p> <p>Con set se configura el valor.</p> |
| Multiples listener. (Configuración en el tnsnames.ora) | <p>(FAILOVER=ON), trata cada dirección hasta que haya una conexión exitosa (pero en orden).</p> <p>(LOAD_BALANCE=ON), toma una dirección aleatoriamente. (Es la mejor para no saturar un listener, de modo que es mas proporcionado la carga de los listener).</p> |
| Verifica los protocolos soportados en el servidor. | <pre>cd \$ORACLE_HOME/bin ./adapters oracle</pre> |
| Prioridad en el sqlnet.ora. | <p>NAMES.DIRECTORY_PATH = (EZCONNECT, TNSNAMES, HOSTNAME). Si no se configura el default es localnaming, Oracle Internet Directory y hostnaming.</p> |
| Especificando un dominio en el sqlnet.ora. | <p>NAMES.DEFAULT_DOMAIN = dominio.</p> |
| Configurar el tnsnames.ora. | <p>Tiene 2 secciones: ADDRESS_LIST (Va el protocolo, host del servidor y el puerto) y CONNECT_DATA (Especifica el nombre del</p> |

| | |
|---------------------------|---|
| | servicio y el tipo de conexión). Y ambos sobre un alias definido. |
| Comunica con el listener. | tnsping servicio ó hostname # (Número de veces que se conectará). |

RECOMENDACIONES

- Si la conexión de red está lento puede fallar la conexión. El error es ORA-12535 y ORA-12547, hay que modificar el parámetro INBOUND_CONNECT_TIMEOUT.
- Revisar si el listener esta generando trace porque podría esto llenar el disco.

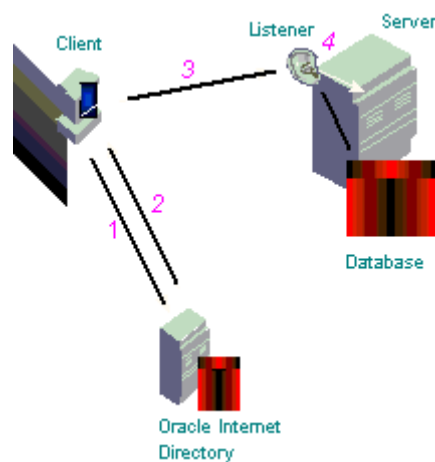
ERRORES COMUNES

ORA-12541 “No listener”, especifica que no hay un listener corriendo en el servidor.
ORA-12198 “Could not find path to destination” y ORA-12203 “Unable to connect fo destination”, especifica que el servicio no corresponde a un servicio valido de base de datos.

ORA-12154 “Cound not resolve connect identifier specified”, especifica que no se puede encontrar el archivo tnsnames.ora o que no se encuentra el nombre del servicio especificado en el tnsnames.ora.

IMÁGENES

- Conexión Directory Naming.



ORACLE SHARED SERVER

CONCEPTOS

- En dedicate mode un cliente se asocia a un server process.
- El PGA = cursor state + user session data + stack space. (Dedicate mode).
- El large pool almacena el cursor state y el user session data, el Large Pool se ubica en el SGA. El stack space solo se queda en el PGA.
- Si no se configura el Large Pool, el UGA pasa a estar en el Shared Pool.
- En el SGA en Shared Server se agregan 2 componentes mas: Request queue y Response queue.
- El memory segment es utilizado para comunicar el cliente y la base de datos.
- El virtual circuit envía y recibe información apropiada por los dispatchers.
- Cuando un cliente se conecta, primero el listener busca el dispatcher menos ocupado y luego envía esta información al cliente que puede conectarse al dispatcher adecuado y este lo hace con el dispatcher adecuado. El PMON registra la conexión asociado al dispatcher en el listener.
- Por defecto cuando se esta en Shared Server, el número de shared servers es 1.
- Share server permite el connection pooling.
- Al añadir nuevos dispatchers estos serán usados por usuarios que inicien sesión no por los que ya estan logeados.
- Otorgando un valor en CIRCUIT ajustamos el total de conexiones de usuarios.
- Aproximadamente cada conexión necesita 1 a 3 mb de ram.
- Con el comando lsnrctl services podemos ver las conexiones por dispatcher, las conexiones rechazadas (debido a una equivocación en el password o se alcanzó el MAX_SHARED_SERVER).
- Para permitir usuarios se conecten via shared server o dedicated; deben tener configurado en el tnsnames.ora el parámetro (SERVER=SHARED ó DEDICATED).

SCRIPTS

| Objetivo | Script |
|--|--|
| Visualizar numero de conexiones, campos: session_current y session_highwater (número de conexiones desde que levanto la base de datos). | v\$license |
| Habilita dispatchers con sus protocolos. Habilita connection pool, el tick es el tiempo que se debe tomar para considerar una conexión ociosa. El valor debe ser multiplicado por 10 min. Ejemplo: TICK = 1 indica 10 minutos. El CONNECTIONS indica la cantidad de conexiones concurrentes que puede soportar y SESSIONS indica cuantas sesiones por dispatcher. Indica la cantidad máxima de dispatchers. | alter system set dispatchers = '(PROTOCOL=protocolo)(DISPATCHERS=#)'; alter system set dispatchers = '(PRO=...)(DIS=#) (POOL=on)(TICK=#)(CONNECTIONS=#)(SESSIONS=#)'; alter system set max_dispatchers = #; |
| Habilita shared servers en el servidor. Indica la cantidad máxima de shared | alter system set shared_servers = #; alter system set max_shared_servers = #; |

| | |
|--|--|
| servers. Especifica el total de sesiones de usuario en el servidor shared. | alter system set shared_server_sessions = #; |
| Especifica la cantidad de circuitos virtuales. | alter system set circuits = #; |
| Muestra los dispatchers en el Linux. Muestra los shared server process en Linux. | ps -ef grep ora_d ps -ef grep ora_s |
| Visualizar los tiempos de trabajo y ocio de los dispatcher en ciento de segundos. Muestra un resumen de los dispatchers por protocolo, # de conexiones, etc. Visualiza ratios de los dispatchers. Muestra información acerca de los request y response queue. Muestra información acerca de la información transmitida del cliente al servidor, etc. Muestra información del numero de request y cantidad de información procesada por los shared servers. Sirve para afinamiento a los Shared Server. Muestra la cantidad de conexiones concurrentes | v\$dispatcher (El campo break indica cuantas veces el usuario cerró la aplicación, antes que reciba los datos solicitado de la base de datos). v\$dispatcher_config v\$dispatcher_rate (El campo cur_msg_rate indica la cantidad de request que ha enviado un dispatcher al Shared Server por segundo, y el campo cur_event_rate indica la cantidad de veces que se ha respondido frente a un request client en minutos). v\$queue (El campo queued indica la cantidad de items que estan esperando en la cola, el campo totalq indica la cantidad de mensajes que han estado en la cola). v\$circuit (El campo queue nos indica si un request esta siendo procesado y el campo saddr lo podemos relacionar a v\$session para saber que usuario esta solicitando un request). v\$shared_sever v\$shared_server_monitor (El campo MAX_CONN indica la cantidad de conexiones que se han alcanzado, MAX_SESS indica la cantidad de sesiones alcanzadas desde que se inicio el servidor, debe compararse con el parametro SHARED_SERVER_SESSIONS). |
| Setea el tamaño del large pool como mínimo. | alter syste set large_pool_size = # |

FORMULAS

- Calculo de DISPATCHERS = CEIL (Máximo numero de *sesiones* concurrentes / Numero de *conexiones* que cada dispatcher maneja (recomendable que el divisor tome el valor de 1000)).
- Calculo del tiempo ocupado de un Share Server:

```
select name, busy+idle as TOTAL_TIME,
case busy+idle when 0 then 0 else busy / (busy+idle) * 100
end as "% BUSSY"
from v$shared_server where status <> 'QUIT'
```
- Calculo del tiempo en cola de los requerimientos (No debe ser mayor a 100 centesimas de segundos, sino aumentar shared_servers):

```
select case totalq when 0 then 0 else
sum(wait/totalq) end as "Tiempo_Espera_Cola (centesimas de seg)"
from v$queue
where type = 'COMMON'
```

- group by totalq
- Calculo del tiempo ocupado de los Dispatchers, si es mayor a 50% debemos aumentar dispatchers:

```

select name, network, (sum(busy) / (sum(busy)+sum(idle)))*100 as "%
BUSY"
from v$dispatcher
group by name, network

```
- Calcular la cantidad de memoria ram que debería tener el large pool:

```

select sum(value)/1024/1024 "MB"
from v$sesstat ss, v$statname st
where st.name = 'session uga memory max'
and ss.statistic# = st.statistic#;

```
- Averiguar tamaño disponible del large pool:

```

select * from v$sgastat where pool = 'large pool';

```

RECOMENDACIONES

- No se debería hacer tareas como reconstrucción de índices, tablas, estadísticas, carga de data en un Shared Server.
- Se debería tener load balancing en el listener.ora si estamos en Shared Server, lo cual permitiría que el listener haga decisiones acerca de que dispatcher esta menos ocupado y enviarlo al cliente.
- Se debería usar v\$session e ir viendo periódicamente el número de conexiones para determinar un correcto número de distpachers.
- Para aplicaciones web utilizar connection pooling.
- Se debería usar Shared Server cuando se utiliza mas de 200 usuarios.

ADMINISTRACIÓN DE USUARIOS

CONCEPTOS

- Oracle ofrece 3 métodos de autenticación: password authentication, external authentication y global authentication.
- El password del usuario por el método password authentication es almacenado en el data dictionary de modo encriptado.
- El Global Authentication permite que Oracle solo verifique que el username sea correcto y la validación se lo deriva a un mecanismo de seguridad avanzada como Kerberos u otros.
- Un perfil sirve para limitar el uso de recursos y además para establecer reglas de passwords.
- El default profile es llamado default.
- Los privilegios permiten que los usuarios puedan acceder a otros objetos que ellos no sean dueños o ejecutar store procedures. También permite al usuario ejecutar operaciones del sistema tal como: conectarse a la base de datos, crear tablas o modificar los parámetros de la base de datos.
- Hay 3 tipos de privilegios: Objetos, Sistema (Creando usuarios, conectarse a la base de datos, alterando la base de datos) y Roles (Objeto + Sistemas agrupados por grupos).
- Cuando se otorga un privilegio de objeto, Oracle registra el que ha garantizado el privilegio y el usuario que se le ha otorgado. Para eliminar un privilegio todos los que han otorgado esos permisos deben ser anulados.
- Cuando se revoca un privilegio de sistema sobre un usuario, todos los privilegios del sistema que este ha otorgado no son revocados a los usuarios porque la base de datos no registra el que otorga el privilegio.
- El comando grant all on objeto otorga permisos totales sobre el objeto.
- Por default el usuario sys tiene los privilegios de SYSDBA y SYSOPER.
- Un usuario que ha sido asignado a un rol hereda un conjunto de privilegios.
- Los profiles sirven para controlar CPU, sesiones, recursos del sistema (memoria, logical reads, etc). Un logical read es una medida de la cantidad de trabajo que la base de datos ejecuta mientras ejecuta un SQL statement.
- Si un usuario es revocado de sus perfiles, él consigue su default perfil.
- Para habilitar la restricción de recursos debemos setear resource_limit a true.
- Restricciones:
 - Prohibiendo que los usuarios no puedan acceder al data dictionary. (alter system set 07_dictionary_accessibility=false;).
 - Revocar los siguientes privilegios a PUBLIC: UTL_TCP, UTL_SMTP, UTL_HTTP, UTL_FILE, DBMS_OBFUSCATION_TOOLKIT y DBMS_CRYPTO.
- Existen algunos roles predefinidos como: CONNECT (Permite a los usuarios acceder a la base de datos), RESOURCE (Permite a los usuarios crear segmentos, ejemplo: tablas; en cualquier tablespace), DBA, EXP_FULL_DATABASE, IMP_FULL_DATABASE, DELETE_CATALOG_ROLE, EXECUTE_CATALOG_ROLE y SELECT_CATALOG_ROLE.
- Existen roles ya creados cuando instalamos Oracle como el CONNECT (permite que los usuarios puedan conectarse a la base de datos), RESOURCE (Permite que el usuario pueda crear segmentos) y DBA (Derecho de ser DBA).

PARAMETROS

| Parámetro | Objetivo |
|--------------------|---|
| remote_os_authent; | Este parámetro permite que clientes se puedan conectar desde una estación de la red y sean validados por su propio sistema operativo. Es recomendable setearlo a false. Si se modifica su valor se debe reiniciar la base de datos. |
| password; | Puede tomar 2 valores SHARED y EXCLUSIVE. El cual cada uno indica si comparte el o no el archivo password file (Archivo donde están incluidos quienes son DBAs) entre varias bases de datos. |
| resource_limit; | Indica si se habilita la restricción de recursos. Puede tomar los valores true o false. |

SCRIPTS

| Objetivo | Script |
|--|--|
| Crear un usuario con password authentication. | <code>create user usuario identified by clave default tablespace nombre_tablespace temporary tablespace nombre_tablespace profile nombre_perfil</code> |
| <u>External authentication:</u> <ol style="list-style-type: none"> 1. Verificando el prefijo. 2. Creando el usuario. 3. Verificando si los usuarios pueden conectarse vía external authentication donde el usuario fue validado por el sistema operativo del cliente. 4. Conectándose desde el localhost. 5. Conectándose desde un cliente. | <code>show parameter os_authent_prefix;</code> <code>create user prefix\$usuario_sistema_operativo identified externally;</code> <code>show parameter remote_os_authent;</code> <code>sqlplus /</code> <code>sqlplus /@tnsnames</code> |
| Seteando un default tablespace para los usuarios. | <code>alter database default tablespace nombre;</code> |
| Modificando el tablespace del usuario. | <code>alter user usuario default tablespace nombre_tablespace;</code> |
| Seteando un temporary tablespace para los usuarios. | <code>alter database default temporary tablespace nombre;</code> |
| Modificando el temporary tablespace del usuario. | <code>alter user usuario temporary tablespace nombre_tablespace;</code> |
| Modificando el perfil de un usuario. | <code>alter user usuario profile nombre_perfil;</code> |
| Eliminando un usuario. | <code>drop user usuario;</code> |
| Eliminando un usuario con sus objetos. | <code>drop user usuario cascade;</code> |
| Otorgando un privilegio de objeto. El with grant option permite que ese usuario pueda seguir otorgando ese privilegio a otros usuarios. | <code>grant (select, insert, delete, alter, execute, etc) on objeto to (public ó nombre_rol ó usuario) with grant option;</code> |
| Muestra los 173 privilegios de sistema. | <code>select * from system_privileg_map;</code> |
| Otorgando un privilegio de sistema. El with admin option permite que ese usuario pueda seguir otorgando ese privilegio a otros usuarios. | <code>grant (alter database, create session, audit system, audit any, restricted session, unlimited tablespace, create user, analyse any, grant any role, sysdba, sysoper y todas las combinaciones de any con</code> |

| | |
|---|--|
| Con el privilegio de sistema create session permitimos que se pueda conectar el usuario a la base de datos. | tables, views, synonym, flashback, index, job etc) to (public ó nombre_rol ó usuario) <i>with admin option</i> ; |
| Devuelve los usuarios con privilegios SYSDBA y SYSOPER. | select * from v\$pwfile_users; |
| Establece un password file para la base de datos. El campo entries indica la cantidad máxima de usuarios con privilegios SYSDBA. | orapwd file=archivo_passwordfile password=password_para_user_sys entries=#; |
| Conectándose como SYSDBA. | sqlplus user/password as sysdba. |
| Creando un rol. Habilitando roles. Deshabilitando roles. <u>Nota 1.</u> - Si un rol fue habilitado en el set y de ahí se ha vuelto a usar el comando set se debe volver a especificar el rol para habilitarlo. <u>Nota 2.</u> - Todos los privilegios con WITH ADMIN OPTION que se han agregado a un rol no se eliminan en cascada. | create role nombre <i>identified by password</i> ; set role nombre <i>identified by password_rol1</i> , rol2 <i>identified by password_rol2</i> , rol3, roln; ó set role nombre all except rol1, roln; set role none; |
| Seteando los roles para un usuario. | alter user usuario default role <i>all except nombre_rol</i> ; <u>Nota:</u> Si hacemos grant nombre_rol to usuario, automáticamente el usuario tendrá habilitado ese rol cuando inicie sesión. |
| Muestra los roles para la session. | select * from session_roles; |
| Muestra los roles disponibles para el usuario. (Existe dba_role_privs). | select * from user_role_privs; |
| Seteando la cuota de un usuario. | alter user usuario quota (# ó unlimited) on nombre_tablespace; |
| Habilita la restricción de recursos. | alter system set resource_limit = true; |
| Creando un perfil. <u>Restricciones de recursos:</u> 1. <u>connect_time</u> : Especifica el tiempo de vida de una conexión. Oracle cierra la conexión y rollback sus transacciones. 2. <u>cpu_per_session</u> : Limita la cantidad de CPU time que puede ser consumida por una sesión. 3. <u>cpu_per_call</u> : Limita la cantidad de cpu que puede ser consumida por una llamada a la base de datos. 4. <u>idle_time</u> : Limita el tiempo de duración entre llamadas a la base de datos. 5. <u>logical_read_per_call</u> : Limita la cantidad de logical reads que puede hacer una llamada de base de datos. 6. <u>logical_read_per_session</u> : Limita la | create profile nombre limit ...; alter profile nombre limit connect_time # ó unlimited; El tiempo está en minutos. alter profile nombre limit cpu_per_session # ó unlimited; El valor está en 1/100 segundos. Este tiempo es por un parse, execute y fetch. alter profile nombre limit cpu_per_call # ó unlimited; El valor está en 1/100 segundos. Este tiempo es por un parse, execute y fetch. alter profile nombre limit idle_time # ó unlimited; El tiempo está en minutos. alter profile nombre limit logical_reads_per_call # ó unlimited; El valor está en unidades de logical reads. alter profile nombre limit logical_reads_session # ó |

| | |
|--|---|
| <p>cantidad de logical reads que puede consumir una sesión.</p> <p>7. <u>private_sga</u>: Limita la cantidad de PGA del usuario.</p> <p>8. <u>session_per_user</u>: Indica la cantidad de sesiones concurrentes por usuario.</p> <p>Todos los valores pueden tomar el valor default.</p> | <p>unlimited; El valor está en unidades de logical reads.</p> <p>alter profile nombre limit private_sga # ó unlimited; El valor está en bytes.</p> <p>alter profile nombre limit session_per_user # ó unlimited; El valor está en unidades de sesiones.</p> |
| Asignando un usuario a su profile. | alter user usuario profile nombre_profile; |
| Mostrando los privilegios que el sysdba a otorgado a los usuarios con sus objetos relacionados. | select * from dba_tab_privs p, dba_objects o where p.owner = o.owner and p.table_name = o.object_name and p.owner = 'SYS' and p.grantee = 'PUBLIC'; |
| Bloquea una cuenta y expira su password. | alter user usuario password expire account lock; Con unlock lo desbloqueamos. |
| <p><u>Implementación de passwords:</u></p> <p>1. Bloqueo de una cuenta si falla n veces consecutivas su logueo.</p> <p>2. Obliga al usuario cambiar su password después de n1 días. Además le permitirá loguearse después de n2 días después del primer logueo exitoso después que se venció.</p> <p>3. Obliga que un password no sea utilizado después de n2 veces donde además no sea utilizando antes de n1 días.</p> <p>4. Incluyendo una función de verificación de password.</p> | <p>create profile nombre limit ... failed_login_attempts n1 password_lock_time n2; *.- Donde n1 es el número de veces consecutivas que falla al loguearse y n2 es el tiempo que se bloquea la cuenta. failed_login_attempts unlimited; (Deshace el bloqueo de cuentas por logueos fallados).</p> <p>password_life_time # ó unlimited password_grace_time # ó unlimited;</p> <p>password_reuse_time n1 ó unlimited password_reuse_max n2;</p> <p>password_verify_function nombre_funcion; *.- La función debe aceptar 3 parámetros de tipo varchar2 (usuario, new_password y old_password) además el owner debe ser sys. Normalmente sirve para validar longitud del password ó complejidad con regular expresion. Para agregar la función a un perfil: create profile nombre limit password_verify_function nombre_funcion;</p> |
| <p>Muestra los primeros 1000 caracteres de un SQL Statement de la base de datos.</p> <p>Muestra un SQL Statement de la base de datos completamente.</p> | <p>v\$sql.</p> <p>v\$sqltext.</p> |
| <p>Muestra los permisos y privilegios sobre objetos.</p> <p>Muestra los permisos y privilegios de sistema.</p> | <p>dba_tab_privs.</p> <p>dba_sys_privs.</p> |
| Indica los perfiles de la base de datos y | dba_profiles. |

| | |
|---|----------------|
| sus límites. | |
| Muestra los roles creados. | dba_roles. |
| Muestra la relación de otorgación de cuotas en cada tablespace por usuario. | dba_ts_quotas. |

RECOMENDACIONES

- Si creamos un script administrativo donde no debemos colocar nuestro password, la mejor alternativa es usar una cuenta externa.
- Nunca se debería dar permisos de un alter system a un usuario externo.
- Setear la variable remote_os_authent a false.
- Todas las cuentas que se crean en el DBCA deberían estar bloqueadas.
- Debemos tener cuidado a los usuarios que tengan los privilegios create any job o procedure y además alter user porque ellos pueden correr programas con privilegios de otros usuarios.

AUDITORIA

CONCEPTOS

- Los registros de auditoria pueden ser registrados en la base de datos o en archivos del sistema operativo para mayor seguridad.
- Por default la tabla de auditoria sys.audit\$ se guarda en el tablespace SYSTEM.
- FGA (Fine Grained Auditing) permite auditar a nivel de campos, incluso por filtros en el campo. Audita a nivel de contenidos de la data.
- Si estamos haciendo un direct path insert en una tabla particionada que esta siendo auditada con FGA causa que se produzcan inserts normales.

PARAMETROS

| Parametro | Objetivo |
|-----------------|---|
| AUDIT_FILE_DEST | Especifica donde se guardarán los registros los archivos de auditoria si se especifico guardar en el sistema operativo. |

SCRIPTS

| Objetivo | Script |
|---|--|
| Especificando el lugar donde se registrará la auditoria. El DB_EXTENDED incluye la sentencia en el campo SQL_TEXT. | alter system set audit_trail=DB ó DB_EXTENDED ó OS scope=spfile; |
| Establece auditoria automatica a los usuarios con privilegios de sysdba, sysoper o al usuario sys. La auditoria no será registrada en sys.aud\$ sino en archivos del sistema operativo. | alter system set audit_sys_operations=false ó true scope=spfile; |
| Muestra los registros de auditoria si se especifico que se guardará los registros en la base de datos. | select * from sys.aud\$; |
| Statement Auditoria Agrega un statement de auditoria. La mayoría se basa en create, drop, alter, grant y revoke. El session indica cuando el usuario se logea. Muestra los statement de auditoria publicados. Examinando la auditoria. Deshabilitando la auditoria. Si no se ingresa whenever automáticamente se desactivan ambos (successful y not successful). Si se esta auditando un usuario determinado y se | audit (table, view, index, function, procedure, package, trigger, user, tablespace, session, sequence, role, profile, not exist, grant (procedure - table - sequence), (select - update - insert - delete - lock) table y system audit)) <i>by user whenever ...</i> select * from dba_stmt_audit_opts; select * from dba_audit_trail; select * from dba_audit_session; select * from dba_audit_statement; select * from dba_audit_object; select * from dba_audit_exists; noaudit (table ó view ...) <i>by user</i> ; noaudit (table ó view ...) <i>by user whenever not successful</i> ; noaudit (table ó view ...) <i>by user whenever successful</i> ; |

| | |
|--|--|
| <p>desea deshabilitar ese criterio de auditoria, hay que especificar el usuario para deshabilitar la auditoria.</p> | |
| <p>Auditando Privilegios Agrega una auditoria de privilegio. La mayoría se basa en create any table, delete any table, etc. Muestra los privilegios de auditoria publicados.</p> | <p>audit (create any table, delete any table, etc) <i>by user by access ó session</i>;</p> <p>select * from dba_priv_audit_opts;</p> |
| <p>Auditando Objetos Agrega una auditoria de objeto. Select, insert, delete, update y execute. Esto es habilitado para todos los usuarios o para ninguno, no se puede auditar para un usuario determinado. Muestra los objetos de auditoria publicados.</p> | <p>audit (select, insert, delete, update ó execute) on objeto <i>by access ó session whenever successful ó not successful</i>;</p> <p>select * from dba_obj_audit_opts; Donde el / en el resultset indicará los successful y los not successful respectivamente. El A indica by access y S indica by session.</p> |
| <p>Limpiando el audit. trail.</p> | <p>delete from sys.audit\$ where timestamp# < sysdate - (# dias);</p> |
| <p>Generando auditoria sobre la tabla sys.audit\$, en caso que alguien lo limpie.</p> | <p>audit all on sys.aud\$ by access;</p> |
| <p>Consultando usuario vs acciones.</p> | <p>select name "action", userid "user", terminal "terminal", timestamp# "time" from sys.audit_actions, sys.aud\$ where action# = action;</p> |
| <p>FGA 1. Creando una política FGA. En el parámetro audit_condition podemos usar AND, OR, etc. Pero no podemos utilizar las funciones sysdate, user, rownum ni usar subqueries. Los handle permiten ejecutar un procedure. Existe un parámetro llamado audit_column_ops el cual puede tomar 2 valores: DBMS_FGA.ALL_COLUMNS (el cual obliga que todos los campos deben ser mencionados para registrarlo en la tabla de auditoria y DBMS_FGA.ANY_COLUMNS. 2. Activando una política. 3. Desactivando una política. 4. Eliminando una política.</p> | <p>EXEC DBMS_FGA.ADD_POLICY(object_schema=>'user',object_name=>'objeto_auditar',policy_name=>'nombre_politica',audit_column=>'campo1',campon',enable=>(TRUE ó FALSE),statement_types=>'select, insert, delete, update',audit_condition=>'campo1>valor1',campon<>valorn',handler_schema=>'owner_procedure',handler_module=>'procedure');</p> <p>EXEC DBMS_FGA.ENABLE_POLICY(object_schema=>'user',object_name=>'objeto_auditado',policy_name=>'nombre_politica');</p> <p>EXEC DBMS_FGA.DISABLE_POLICY(object_schema=>'user',object_name=>'objeto_auditado',policy_name=>'nombre_politica');</p> <p>EXEC DBMS_FGA.DROP_POLICY(object_schema=>'user',object_name=>'objeto_auditado',policy_name=>'</p> |

| | |
|--|---|
| 5. Consultando las políticas registradas. | >'nombre_politica'); |
| 6. Consultando los registros FGA. | select * from dba_audit_policies; |
| 7. Limpiando los registros FGA. | select * from dba_fga_audit_trail; truncate table sys.fga_log\$; |
| Muestra las acciones disponibles. | select * from sys.audit_actions; |
| Vista de los registros FGA + registros de auditoría comunes. | dba_common_audit_trail; |

RECOMENDACIONES

- Mover la tabla sys.aud\$ a otro tablespace.
- Auditar la tabla sys.aud\$.

PL/SQL

CONCEPTOS

- MTTR = Tiempo para un recovery, MTBF = El tiempo de diferencia entre una falla y otra.
- PL/SQL maneja secuencias, condiciones de control, loops, excepciones, colecciones, métodos, herencia, etc.
- Un paquete tiene registros, cursores, variables, constantes, procedures y funciones.
- Cuando se garantiza permisos sobre un package se entregan permisos sobre sus objetos por default. No se puede garantizar permisos a un exclusivo objeto de un package.
- DML triggers no pueden ser especificados en SYS-owned objects.
- Los eventos de un trigger pueden ser por DDL, DML y BD.
- Los directorios no son objetos de esquemas como las tablas etc. Son perfiles o roles los cuales tiene como dueño a la base de datos.
- Cuando un MERGE se ejecuta hace lo siguiente:
 - Parse: Revisa sintaxis, privilegios y plan de ejecución.
 - Bind: Asigna valores a algunas variables.
 - Execute: Es recibida del database buffer cache cuando se ejecuta un statement.

- Cuando hay un ordenamiento que excede el tamaño del TEMPORARY tablespace ó un bloque de transacciones que genera más undo segments de los que hay ó un create table as select * from tabla ó un create index ó mover una tabla, importar o exportar puede existir problemas de resumable space management, para evitar este problema debemos crear un trigger que se ejecute despues de un **after suspend on database** y ejecutar **dbms_resumable.set_timeout(# en segundos)** y la sesión debe tener un **alter session enable resumable**. Buena práctica es hacerlo desde un trigger que se ejecute en el **after logon on database** y llame al **execute immediate 'alter session enable resumable'**.

PARAMETROS

| Parámetros | Objetivo |
|---------------------|---|
| ERROR | Muestra el error producido al crear un PL/SQL. |
| PLSQL_WARNING | Ayuda a identificar problemas en el desarrollo. Debe estar seteado a 'DISABLE:ALL'. |
| PLSQL_DEBUG | Fuerza a todos los codigos a ser interpretados y se incluye extra información de debugging. Debe ser seteado a FALSE. |
| PLSQL_OPTIMIZE_MODE | Permite compilaciones rápidas. Por default tiene el valor de 1, debemos pasarlo al valor de 2. |
| PLSQL_CODE_TYPE | Si lo seteamos a NATIVE debemos tener el compilador C++ y recrear los procedures y funciones y setear la variable PLSQL_NATIVE_LIBRARY_DIR. |

SCRIPTS

| Objetivo | Script |
|--|--|
| Multiples inserts, es más rápido que estar ingresando 2 inserts. | insert first when condicion1 then into tabla1 values (campos_tabla2 o constantes) when condicionn then into tabla1 values (campos_tabla2 o constantes) select * from tabla2; |
| Muestra el rowid de una fila. | select rowid from tabla; |
| Muestra todos los procedures de la base de datos. | dba_sources. |
| Muestra todos los triggers de la base de datos. | dba_triggers. |
| Devuelve un valor deseado de una fecha. | to_date(variable_fecha,'valor'), donde el valor puede ser year, day, month, etc. |
| Funciones. <u>Nota 1.</u> - Si la función no tiene parámetros no incluir paréntesis vacíos. <u>Nota 2.</u> - Para asignar un valor a una variable, debe usarse :=. <u>Nota 3.</u> - Para recoger los valores de un select debemos poner into var1, var2, etc entre el select y el from. | CREATE OR REPLACE FUNCTION nombre (<i>par1 tipo_dato, parn tipo_dato</i>) return tipo_dato IS <i>var1 tipo_dato default valor</i> BEGIN ... acciones ... return valor; END; / |
| Procedures. | CREATE OR REPLACE PROCEDURE nombre (<i>par1 tipo_dato, parn dato</i>) IS <i>var1 tipo_dato default valor</i> |

| | |
|--|---|
| | BEGIN END; / |
| Packages. Cabeceras. Body. | CREATE OR REPLACE PACKAGE nombre IS functions ... procedures ... END; CREATE OR REPLACE PACKAGE BODY nombre IS functions ... procedures ... END; / |
| Compila un package. | ALTER PACKAGE nombre COMPILE BODY; |
| Triggers DML. <u>Nota 1.</u> - INSERTING, UPDATING, DELETING son constantes para identificar el tipo de operación que se esta haciendo. <u>Nota 2.</u> - Podemos usar OR para referirnos a varias operaciones en un mismo trigger, ejemplo: INSERT OR DELETE etc. <u>Nota 3.</u> - El nuevo valor es :NEW y el viejo valor es :OLD. Para referirnos a ellos debemos especificar su campo, ejemplo: :NEW.campo:=valor; | CREATE OR REPLACE TRIGGER nombre (BEFORE ó AFTER) (INSERT, DELETE, UPDATE OF nombre_campo) ON tabla FOR EACH ROW BEGIN ... END; / |
| Triggers DDL. Las acciones pueden ser: ALTER, CREATE, DROP, TRUNCATE, COMMENTS, ANALYZE Ó ASSOCIATE STATISTICS, DISASSOCIATE STATISTICS, NOAUDIT, AUDIT, GRANT, REVOKE Y RENAME. | CREATE OR REPLACE TRIGGER nombre (BEFORE ó AFTER) (ACCION) on (DATABASE ó nombre_schema.SCHEMA) BEGIN ... END; / |
| Triggers BD. Las acciones pueden ser: AFTER LOGON, BEFORE LOGOFF, AFTER STARTUP, BEFORE SHUTDOWN, AFTER SERVERERROR y BEFORE ó AFTER SUSPEND. | CREATE OR REPLACE TRIGGER nombre (ACCION) ON DATABASE BEGIN END; / |
| Habilita o deshabilita un trigger. | ALTER TRIGGER nombre DISABLE ó ENABLE; ALTER TABLE nombre_tabla ENABLE ó DISABLE ALL TRIGGERS; |
| Condicionales | IF Condicion THEN ... ELSE ... END IF. CASE Valor WHEN valor1_comparar THEN RETURN valor1; WHEN valorn_comparar THEN RETURN valorn; ELSE RETURN valor; END CASE; |
| Bucles. Condición dentro del bucle. | LOOP ... END LOOP; Para salir del LOOP usar EXIT. |

| | |
|---|---|
| Condición al comenzar el bucle. | WHILE Condicion LOOP ... END LOOP. FOR Contador IN valor_min..valor_max LOOP END LOOP. |
| Muestra valores en la consola. | exec dbms_output.put_line('valor'); |
| Crea un statement en tiempo de ejecución. Lo unico que no se puede hacer es devolver multiples registros como un select. | set serveroutput on begin exec immediate 'statement_generado'; end; / |
| Lanza un error. | raise_application_error(#, 'valor'); |
| Devuelve true o false si el error que se ha generado en el servidor es un código determinado. | is_servererror(#); |
| Visualizar las modificaciones de los objetos. | dba_objects (En los campos: created, last_ddl_time podemos visualizar los cambios hechos en los objetos). |
| Crea un directorio. | create directory nombre as 'ruta'; |
| Otorga permisos sobre un directorio. | grant read, write on directory directory nombre_directorio to public or nombre_usuario, etc. |
| Muestra las sesiones que tienen el resumable habilitado. | select * from dba_resumable; |
| Visualiza el codigo fuente de los triggers. | dba_triggers. |
| Visualiza el codigo fuente de los procedures, triggers, etc. | dba_source. |

RECOMENDACIONES

- Para conseguir una mejor performance en las aplicaciones debemos tener
seteado lo siguiente:
 - PLSQL_CODE_TYPE = NATIVE.
 - PLSQL_DEBUG = FALSE.
 - PLSQL_OPTIMIZE_MODE = 2.
 - PLSQL_WARNING = DISABLE:ALL.

SQL LOADER

CONCEPTOS

- SQL Loader Direct es más rápido que el convencional. El convencional pasa a la memoria y utiliza undo data porque utiliza inserts convencionales. Con el Direct utiliza APIS el cual permiten grabar directamente en los datafiles no pasando por el buffer cache. En conclusión el directo no genera redo, no lanza triggers y va directo a disco. El problema de usar Direct es que bloqueamos la tabla.
Las restricciones que tiene el SQL Loader Direct son las siguientes:
 - No trabaja sobre tablas particionadas.
 - Los constraint de foreign key son deshabilitados durante la carga.
 - Los constraint de unique no son considerados y son dejados invalid después de la carga.
 - No puede realizarse si hay transacciones activas sobre la tabla.
- El archivo bad file indica cuales son los registros que han fallado al realizarse la delimitación.
- El archivo log file indica cuales son los registros que no han cumplido con las reglas de validación.
- Una manera de conseguir mayor performance en la carga sin usar Loader Direct es incrementar el bind size (parametro bindsize=#, por default es 256 KB, el cual indica el tamaño del arreglo en bytes), lo cual provoca que no haya tantos viajes por la red.

PASOS

- Crear un archivo control con extensión ctl.
- Para ejecutar el sqlldr se usa el siguiente comando:
C:> sqlldr user/password archivo.ctl errors=#
Indicando un número en errors estamos estableciendo la cantidad de errores tolerados, este parámetro es opcional.
- El contenido del archivo control:

```
OPTIONS (DIRECT=FALSE ó TRUE)
CONTINUE LOAD DATA
INFILE * ó archivo_fuente_datos
DISCARDMAX #
    INTO TABLE tabla1 (TRUNCATE, INSERT, APPEND ó DELETE)
WHEN (#) ='valor' (or ó and) ...
TRAILING NULLCOLS
    (
        CAMPO POSITION(X:Y) (CHAR, INTERGER EXTERNAL ó DECIMAL
        EXTERNAL),
        ...
    )
    ...
    INTO TABLE tablan (TRUNCATE, INSERT, APPEND ó DELETE)
WHEN CAMPO! ='valor' (or ó and) ...
FIELDS TERMINATED BY ","
TRAILING NULLCOLS
    (
        CAMPO (CHAR, INTERGER EXTERNAL ó DECIMAL EXTERNAL) NULLIF
        CAMPO=BLANKS
```

)
BEGINDATA
valora1,valorb2
valoran,valorbn

OPTIONS.- Por default es FALSE. Indica si vamos a utilizar Direct Path Loading si es TRUE, si es FALSE indica que vamos a utilizar el convencional.

CONTINUE.- Se utiliza cuando ya ha habido una carga de datos antes y se desea continuar con la carga que ha sido interrumpida.

INFILE.- Indica donde se encuentran los datos. Si ponemos * indica que los datos están dentro del mismo archivo y utiliza la sección BEGINDATA para tomar los datos, en caso contrario lee los datos en el archivo_fuente_datos.

DISCARDMAX.- Indica hasta cuantos registros que no cumplan con las reglas aplicadas serán tolerados para no parar la carga, por default si deseamos que no pare la carga por n registros que no cumplan las condiciones no debemos escribir la sección DISCARDMAX. Opcional.

TRUNCATE, INSERT, APPEND ó DELETE.- Truncate limpiará la tabla antes de comenzar la carga. Para utilizar insert la tabla debe estar vacía.

WHEN(#) = 'valor'.- Filtramos datos, en el cual indica que las filas cuya posición # tenga el valor indicado, ejemplo WHEN(2)='a' indica que solo los registros cuyas filas tengan en la posición 2 el valor de a ingresa en la carga. Podemos filtrar también por campos. Opcional.

TRAILING NULLCOLS.- Indica que toda columna que sea NULL la inserte en el registro como NULLS. Lo mismo provoca si agregamos la siguiente línea a los campos NULLIF CAMPO=BLANKS.

DELIMITACION.- La delimitación de campos se puede hacer por rangos es decir posición 1 al 10 y el siguiente a la posición 11 al 35 ó lo podemos hacer por separación por comas, etc, ejemplo: FIELDS TERMINATED BY ",".

TABLAS EXTERNAS

Pasos:

- Crear un directorio:
CREATE DIRECTORY BDUMP AS 'ruta_directorio';
- Crear una tabla:
CREATE TABLE nombre
(
CAMPO VARCHAR2(#)
)
ORGANIZATION EXTERNAL
(
TYPE ORACLE_LOADER
DEFAULT DIRECTORY BDUMP
ACCESS PARAMETERS
(
RECORDS DELIMITED BY NEWLINE
BADFILE 'TABLE nombre'
LOGFILE 'TABLE nombre'
FIELDS TERMINATED BY ','
)
LOCATION ('nombre_archivo_cargar')
) REJECT LIMIT UNLIMITED;

*.- Donde la delimitacion puede ser por la , o por otro carácter.

DATA PUMP

CONCEPTOS

- En caso de los índices al exportar solo se exporta definición y al importar se reconstruye los índices.
- Para exportar una base de datos o exportar un esquema que no sea el nuestro o exportar tablespaces debemos tener los privilegios de EXP_FULL_DATABASE rol y para importar IMP_FULL_DATABASE rol.
- Los archivos de exportación residen en el servidor.
- Datapump permite una transferencia de datos entre base de datos de modo rápido.
- Debemos tener un objeto directorio como requisito para usar data pump.

USOS

EXPORTAR

- Exportación de una base de datos.

```
expdp user/password full=Y directory=directorio_object  
dumpfile=nombre_archivo%U.dmp filesize # parallel=# logfile=nombre_archivo.log
```

- Exportación de schemas.

```
expdp user/password schemas=esquema1, ...,esqueman  
directory=directorio_object dumpfile=nombre_archivo%U.dmp filesize # parallel=#  
logfile=nombre_archivo.log
```

- Exportación de tablas.

```
expdp user/password tables=tabla1,...,tablan  
content=data_only ó metadata_only ó ALL  
directory=directorio_object dumpfile=nombre_archivo%U.dmp filesize # parallel=#  
logfile=nombre_archivo.log
```

- Exportación de tablespaces.

```
expdp user/password tablespaces=tablespace1, ...,tablespacen  
directory=directorio_object dumpfile=nombre_archivo%U.dmp filesize #  
parallel=# logfile=nombre_archivo.log
```

- Opciones adicionales.

- Leyendo de un archivo todos los parámetros para exportar:
expdp user/password parfile=archivo
- Usando la ayuda.
expdp help=Y
- No generando el archivo log y visualizandolo en la pantalla.
expdp ... nologfile=Y
- Se acceda a una base de datos remota para la exportación.
expdp network_link=dblink
- Setear un nombre a la exportación.

- expdp job_name=nombre
- Seteamos cierta cantidad de procesos para ejecutar la exportación, el default es 1.
- expdp parallel=#

IMPORTAR

- Importación de una base de datos.

impdp user/password full=Y directory=directorio_object dumpfile=nombre_archivo nologfile=Y

- Importación de schemas.

impdp user/password schemas=esquema1, ...,esqueman
 remap_schema="user_origen:user_destino" content=data_only ó
 metadata_only ó ALL directory=directorio_object dumpfile=nombre_archivo
 nologfile=N

- Importación de tablas.

impdp user/password tables=tabla1,...,tablan content=data_only ó
 metadata_only ó ALL remap_schema="user_origen:user_destino"
 directory=directorio_object dumpfile=nombre_archivo nologfile=N

- Importación de tablespaces.

impdp user/password tablespaces=tablespace1, ...,tablespacen
 directory=directorio_object dumpfile=nombre_archivo nologfile=N

- Opciones adicionales.

- Content: Podemos importar data, metadata o ambos.
 impdp user/password content=DATA_ONLY ó METADATA_ONLY ó ALL.
- Usando la ayuda.
 impdp help=Y
- No generando log y visualizandolo en la pantalla.
 impdp ... nologfile=Y
- Se acceda a una base de datos remota para la importación.
 impdp network_link=dblink
- Setear un nombre a la importación.
 impdp job_name=nombre
- Seteamos cierta cantidad de procesos para ejecutar la importación, el default es 1.
 impdp parallel=#
- Consiguiendo los DDL statment.
 impdp sqlfile = archivo. (El parallel debe estar seteado a 1).
- Cambiando de schema, datafile y tablespace. Esto transfiere los objetos de un lado a otro.
 impdp remap_(schema ó datafile ó tablespace)="user_origen:user_destino"
- Reglas de inclusión y exclusión.
 impdp include=(PROCEDURE ó TABLE ó VIEW etc):"LIKE ó = 'valor%'"

Si no se pone el filtro (a partir del :), se considera todas las tablas ó vistas etc que se desea importar. Se puede poner varios includes, uno debajo del otro.

- Reusando datafiles.
impdp reuse_datafiles=Y

Nota.- La vista dba_datapump_jobs muestra el monitoreo de los jobs datapump.

UNDO TABLESPACE - CONSISTENCIA

CONCEPTOS

- El undo data permite hacer un rollback a la transacción del usuario.
- El Undo Adviser recoge estadísticas para recomendar el tamaño del undo tablespace y con tensión del undo tablespace.
- Dependiendo del undo puede llegar a usarse el flashback query.
- Los undo segments son llamados también rollback segments.
- Los undo segments crecen y se encojen automáticamente.
- Toda transacción es asignada a un solo undo segment, si le falta espacio se ubica un nuevo extent al undo segment. Puede darse el caso que varias transacciones utilicen el mismo extent.
- El owner de los undo segments es el usuario SYS.
- Para recuperar un undo tablespace debe estar en modo MOUNT y lo hace a través de los online redo log ó de los archived redo log.
- Los usuarios SYS o SYSTEM tiene un segmento especial reservado, con el código USN = 0. También utilizan este segmento especial aquellos cambios que se realicen sobre el tablespace SYSTEM.
- En el undo data va el viejo valor de la data de una tabla o un índice.
- Genera undo el delete, insert, update y el merge.
- Cuando se hace rollback se liberan los bloqueos de la fila.
- Los inserts usan poco undo segment porque se almacena un puntero a la nueva fila y la elimina si la transacción es rollback.
- Al modificar un usuario con alter user también genera transacción porque modifica las tablas de diccionario.
- Cuando se cambia de auto o manual el UNDO_MANAGEMENT debe ser reiniciado la instancia.
- Las alertas generadas por insuficiencia de espacio por el undo_retention es lanzada 1 vez cada 24 horas.
- El tablespace undo esta por default NOGUARANTEE.
- Cuando se cambia de undo tablespace se mantiene aún el viejo undo tablespace hasta que este no tenga transacciones pendientes. Las nuevas transacciones utilizarán el nuevo undo tablespace.
- Si no especificamos el parametro UNDO_TABLESPACE y estamos seteando el parámetro UNDO_MANAGEMENT = AUTO, Oracle creará un tablespace llamado SYS_UNDOTS (Esto cuando se creó la base de datos).

SCRIPTS

| Objetivo | Script |
|---|--|
| Mostrar los undo segments. | v\$rollname. |
| Indicar un nombre a una transacción. | set transaction name 'nombre'; ... (updates, delete, etc). |
| Mostrar la relación entre una transacción y el undo segment asignado. | select xid, status, start_time, xidusn as seg_num, r.name as seg_name from v\$transaction t join v\$rollname r on t.xidusn = r.usn where t.name = 'nombre'; |
| Prohibir operaciones de DML en una sesión. | set transaction read only; |
| Configurar el undo tablespace. | UNDO_MANAGEMENT = AUTO; Para este paso se debe reiniciar la instancia (scope = spfile). UNDO_TABLESPACE = nombre; Especifica el |

| | |
|---|---|
| | undo tablespace. UNDO_RETENTION = #; El valor de 0 indica un tiempo asignado por Oracle. |
| Muestra el trabajo del tablespace undo cada 10 minutos. Donde: <ul style="list-style-type: none"> • Undoblks: Representa el total de undo blocks consumidos. • Maxquerylen: Representa la cantidad de segundos del mas largo query en un intervalo de tiempo. (Se cuenta desde que se abre el cursor hasta el último fetch). | select to_char(begin_time,'yyyy-mm-dd hh24:mi:ss') as starttime, to_char(end_time,'yyyy-mm-dd hh24:mi:ss') as endtime, undoblks, maxquerylen from v\$undostat; Esto es solo utilizado para estimar el tamaño del undo tablespace. |
| Muestra el tamaño del undo tablespace. Viene ser igual a la suma de bytes de los datafiles que conforman el tablespace. | select sum(a.bytes)/1024/1024 "MB UNDO_SIZE" from v\$datafile a, v\$tablespace b, dba_tablespaces c where c.contents = 'UNDO' and c.status = 'ONLINE' and b.name = c.tablespace_name and a.ts# = b.ts#; |
| Poner un tablespace en guarantee. Poner un tablespace en noguarantee. | alter tablespace nombre retention guarantee; alter tablespace nombre retention noguarantee; |
| Visualizar si el tablespace esta en guarantee o no. | select tablespace_name, contents, block_size, retention from dba_tablespaces; |

FORMULAS

- Para estimar el tamaño necesario del undo tablespace:

$$\text{size} = \text{UR} (\# \text{ de undo retention, ejemplo: } 1000) * \text{UPS} (\text{La máxima cantidad de undo blocks utilizado, sale en consecuencia de utilizar la vista v\$undostat en el campo undoblks}) * \text{tamaño del bloque en el tablespace (Ejemplo: 8192 bytes es decir 8k, se puede conseguir de: select tablespace_name block_size from dba_tablespaces)}.$$

Al tamaño obtenido agregarle 10% o 20% más en caso de subidas de piques en el undo usado.

Nota: La respuesta esta en bytes si se ingresa el tamaño del bloque en bytes, si se ingresa en kb la respuesta esta en kb.

RECOMENDACIONES

- Los datafiles se deberian tener en autoextend hasta un tiempo y una vez que ya es constante el tamaño se le debería quitar lo la opción de autoextend.

BLOQUEOS

CONCEPTOS

- En Oracle se puede bloquear en una o varias filas o la tabla completa.
- Cuando se genera un deadlock, esto es escrito en el alert de la base de datos y además se genera un archivo trace en el udump detallando el error. Además la primera sesión que inicio el bloque de transacciones, su ultima transacción es anulada y la otra sesión queda en espera.
- Los bloqueos pueden ser exclusivos el cual no permite que la tabla sea modificada y share el cual permite que otros usuarios puedan bloquear partes del mismo recurso (Ejemplo: cuando hacemos un insert solo bloqueamos una fila pero otro usuario podria bloquear otra fila).
- Las vistas de referencia a bloqueos son v\$lock y v\$locked_object.
- El comando nowait permite que una sesión no se quede esperando un tiempo infinito al solicitar un recurso bloqueado. Ejemplo: select * from tabla for upate nowait;
- Para cancelar un bloqueo se hace mediante un rollback o commit.
- En el Enterprise Manager se puede visualizar las sesiones que bloquean recursos. Lo mismo lo podemos conseguir por la vista v\$locked_object pero no muestra quienes están esperando por recursos.

SCRIPTS

| Objetivo | Script |
|---|---|
| Permite indicar que usuarios estan teniendo problemas con bloqueo de recursos. Los que no aparecen con espacios en blanco indica que ellos son los que causan los problemas de bloqueo en la consulta presentada. | <pre>select lpad(' ',decode(l.xidusn,0,3,0)) l.oracle_username "Usuario", o.owner, o.object_name, o.object_type from v\$locked_object l, dba_objects o where l.object_id = o.object_id order by o.object_id, 1 desc;</pre> |
| Permite mostrar cuales son las sesiones en el servidor de base de datos. | <pre>select sid,serial#, username, terminal, status from v\$session where username is not null;</pre> |
| Aniquila una sesión. | <pre>alter system kill session '#sid,#serial';</pre> |
| Seleccionar registros y bloquearlos. | <pre>select * from tabla for update;</pre> |
| Bloquea una tabla en modo explícito, donde el modo puede ser: Exclusive, share, row share, row exclusive y share row exclusive. | <pre>lock table tabla in nombre_modo mode;</pre> |

TUNING

CONCEPTOS

- Existen 2 procesos MMON y el MMNL (Memory Monitor Light) recolectan estadísticas directamente del SGA.
- El MMON recolecta estadísticas cada 60 minutos (no genera estadísticas) y estas perduran una semana por default. El recolecta información de las vistas de diccionario, vistas dinámicas, vistas de performance y la almacena en tablas denominadas AWR que se ubican en el tablespace SYSAUX y su owner es SYSMAN.
- El ADDM analiza el snapshot y lo compara con los 2 anteriores. Identifica problemas como: CPU, cuellos de botella de I/O, PL/SQL que demandan gran trabajo, bloqueos y la utilización del SGA. La meta del ADDM es minimizar el DBTime. El ADDM publica su resumen en el Database Control.
- El DBTime esta compuesto por el CPU time y el tiempo en espera. El cual indica el tiempo gastado usando recursos del CPU y el tiempo en espera para acceder a algun recurso. Cuando incrementa el dbtime indica que los usuarios necesitan utilizar más recursos ó que los usuarios estan esperando por aquellos recursos.
- Los advisors son los siguientes: SQL Tuning, SQL Access Advisor, Memory Advisor, Mean Time to Reover Advisor, Segment Advisor y el Undo Management Advisor.
- El SQL Tuning Advisor permite afinar los SQL statement que estan contribuyendo a incrementar el DBTime.
- El SQL Access Advisor provee otras maneras de reducir la cantidad de I/O o de CPU o de wait time, etc.
- Ambos el SQL Tuning Advisor y el SQL Access no trabajan ni avisan sobre indices fragmentados y sobre los queries que trabajan sobre ellos. Esto solo lo hace el Segment Advisor.
- Memory Advisor se encarga de aconsejar sobre el SGA y el PGA. En la pantalla del Control Manager en el Memory Advisor hay un botón llamado Automatic Shared Memory Management (Disable – Enable), esto permite que ADDM puede automaticamente ajustar.
- MTTR (Mean Time to Recover) Advisor aconseja sobre el tiempo de un instance recoery. El FAST_START_MTTR_TARGET no debe estar seteado a cero.
- Segment Advisor, identifica los segmentos que pueden ser encojidos. Existe 2 opciones en el Segment Advisor para encojer: Compact Segments (Encoje pero no libera el espacio encojido) y Compac Segments and Release Space (esto si lo hace). Detecta problemas de fragmentación.
- El Undo Advisor nos permite que los usuarios siempre encuentre undo segments disponibles para sus transacciones, que la base de datos siempre consigue read-consistent y que la base de datos no genere innecesarios de I/O. Detecta además problemas de contención en el UNDO tablespace.
- El ADDM envía alertas notificando problemas para acciones correctivas.
- Las metricas son medidas que son almacenadas en el AWR. Al generar estadísticas, el ADDM compara las metricas con los baseline. Podemos crear nuestros propios baseline de un día óptimo.
- El ADDM detecta:
 - Lock contention.
 - Cuellos de botellas de CPU.
 - Altos tiempos de ejecución de los PL/SQL y Java.
 - I/O capacidad.

- Altos SQL statement y sus ejecuciones.
- Muy seguidos check points.
- Mal dimensionado las estructuras de memorias.
- Aproximadamente son más de 350 vistas dinamicas de performance.
- Al tener estadísticas permite generar eficientes planes de ejecución para cada SQL statement.
- Las estadísticas las podemos poner en shedule desde el Database Control (Mantenimiento\Recompilar Estadísticas).
- Cuando recompilamos estadísticas tenemos 2 opciones: Compute statistics (Genera estadísticas a todo el objeto) ó estimate statistics (Toma una muestra que le indiquemos o una muestra que Oracle determine aceptable para generar estadísticas).
- Una metrica importante es throughput que indica el rendimiento de procesamiento ó la cantidad de trabajo que el sistema puede ejecutar en un determinado tiempo (es considerado para la escalabilidad del sistema) y otra metrica es importa es el response time el cual indica el tiempo que demora un request de un usuario en cumplirlo. Estas métricas se ven en el Database Control en la sección Todas las métricas.
- Los jobs scope pueden ser limited scope (tiene menor impacto en el sistema) y comprehensive scope.
- Al crear una base de datos con DBCA esta configurada para recolectar estadísticas de tablas y estadísticas entre 10 pm y 6 am.
- Si especificamos un warning threshold de 200% en una métrica, significa que notificará cuando la métrica sea el doble de la línea base.

PARAMETROS

| Parámetro | Objetivo |
|---|---|
| STATISTICS_LEVEL = (BASIC, TYPICAL, ALL). ALL se utiliza cuando se pasa de desarrollo a producción y cuando ya se tiene la base de datos estable. | Especifica el nivel de recolección de las estadísticas. |
| LOCK_SGA | Evitamos usar memoria SWAP. Debemos tener cuidado porque podemos conseguir el siguiente error: ORA-27102: out of memory, ahí debemos disminuir el LARGE_POOL_SIZE. |
| BACKGROUND_DUMP_DEST y USER_DUMP_DEST | Muestra la ruta donde se guardan los traces generados por el usuario o por los server process. Usualmente los traces se utilizan encontrar problemas de performance y encontrar cuellos de botella. |

SCRIPTS

| Objetivo | Script |
|---|---|
| Especifica un nuevo tiempo de recolección y de retención de una foto. | EXEC DBMS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS(INTERVAL=>#,RETENTION=>#); Los números están expresados en minutos. Si se ingresa un valor con null se mantiene el valor por defecto. |
| Muestra la configuración del AWR. | select snap_interval, retention from dba_hist_wr_control; |
| Muestra la lista de los snapshots | select snap_id, begin_interval_time, |

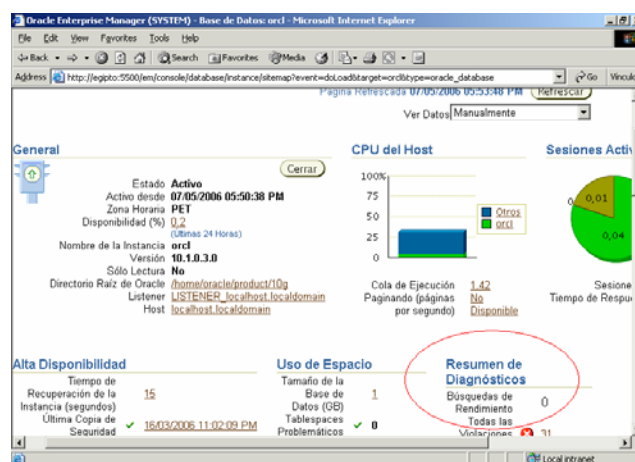
| | |
|--|---|
| generados. Muestra información de los baseline. | end_interval_time from dba_hist_snapshot; dba_hist_baseline. |
| Genera un reporte del ADDM en un intervalo de snapshots. | sqlplus user/password @?/rdbms/admin/addmrpt.sql |
| Genera un reporte sin usar el script addmrpt.sql. Utiliza 2 intervalos de snapshots. | 1. EXEC DBMS_ADVISOR.CREATE_TASK(ADVISOR_NAME=>'ADDM',TASK_NAME=>'nombre_tarea',TASK_DESC=>'descripcion'); 2. EXEC DBMS_ADVISOR.RESET_TASK(TASK_NAME=>'nombre_tarea'); 3. EXEC DBMS_ADVISOR.SET_TASK_PARAMETER(TASK_NAME=>'nombre_tarea',PARAMETER=>'START_SNAPSHOT',VALUE=>#); # del snapshot inicial. 4. EXEC DBMS_ADVISOR.SET_TASK_PARAMETER(TASK_NAME=>'nombre_tarea',PARAMETER=>'END_SNAPSHOT',VALUE=>#); # del snapshot final. 5. EXEC DBMS_ADVISOR.EXECUTE_TASK(TASK_NAME=>'nombre_tarea'); 6. SELECT DBMS_ADVISOR.GET_TASK_REPORT('nombre_tarea') FROM DUAL; |
| Tomar una foto. Elimina un intervalo de fotos. | EXEC DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT; EXEC DBMS_WORKLOAD_REPOSITORY.DROP_SNAPSHOT_RANGE(LOW_SNAP_ID=>#,HIGH_SNAP_ID=>#); |
| Crear un baseline. Elimina un baseline. | BEGIN DBMS_WORKLOAD_REPOSITORY.CREATE_BASELINE(START_SNAP_ID=>#,END_SNAP_ID=>#,BASELINE_NAME=>'nombre'); END; BEGIN DBMS_WORKLOAD_REPOSITORY.DROP_BASELINE(BASELINE_NAME=>'nombre',CASCADE=>FALSE); END; (Si es TRUE elimina tambien los snapshots). |
| Genera estadísticas manualmente. (DBMS_STATS). | EXEC DBMS_STATS.GATHER_SCHEMA ó DATABASE ó SYSTEM ó TABLE ó INDEX_STATS('owner','nombre_objeto',OPTIONS=>'GATHER AUTO'); Con la opción GATHER AUTO solo recolecta estadísticas a los que necesitan, es el default y no es para todas las opciones como TABLE, INDEX, etc. Existe el parámetro ESTIMATE_PERCENT = # el cual nos |

| | |
|---|---|
| | permite no analizar todo el objeto. |
| Genera estadísticas manualmente. (ANALYZE). | ANALYZE TABLE ó INDEX nombre COMPUTE STATISTICS; ANALYZE TABLE ó INDEX nombre ESTIMATE STATISTICS SAMPLE # ROWS; ANALYZE TABLE ó INDEX nombre ESTIMATE STATISTICS SAMPLE # PERCENT; |
| Vistas de performance. | dba_advisor_findings (Muestra los problemas encontrados para afinar, tomar el campo task_id, type, message). dba_advisor_objects (Describe los objetos que están involucrados por el task_id, tomar los campos attr). dba_advisor_recommendations (Describe los cambios que se deben realizar por el task_id, tomar los cambios attr). dba_advisor_rationale (Describe la explicación de porque se recomienda los cambios de un task_id, tomar el campo message). |
| Performance de un query. | 1. DECLARE l_sql VARCHAR2(500); BEGIN l_sql:=DBMS_SQLTUNE.create_tuning_task (sql_text=>'qry',user_name=>'usuario', scope=>DBMS_SQLTUNE.scope_comprehensive, time_limit=>#,task_name=>'nombre', description=>'descripcion'); END; 2. exec dbms_sqltune.execute_tuning_task ('nombre'); 3. set serveroutput on size 999999 set long 999999 select dbms_sqltune.report_tuning_task('nombre') from dual; 4. exec dbms_sqltune.drop_tuning_task ('nombre'); |
| Especifica que Oracle puede mover los segmentos y borrarlos y agregarlos en pro de dejar el espacio libre al final y además deja habil para actualizar el water mark. | alter table nombre enable row movement; |
| Encoje una tabla. | alter table nombre shrink space <i>cascade</i> ; (Requisito, la tabla debe estar con enable row movement). El cascade se usa para las dependencias de la tabla. |
| Vistas importantes. | V\$SGASTAT = Información del SGA. V\$EVENT y V\$SYSTEM_EVENT = Muestra los sucesos de la base de datos. V\$SYSSTAT = Muestra información de todos los aspectos de la base de datos. (La mas principal). DBA_TABLES, DBA_INDEXES, DBA_SEGMENTS, DBA_DATA_FILES, DBA_OBJECTS (Importante) son vistas donde recojemos información adecuada. En el caso de las tablas se actualiza la fecha analizada, la |

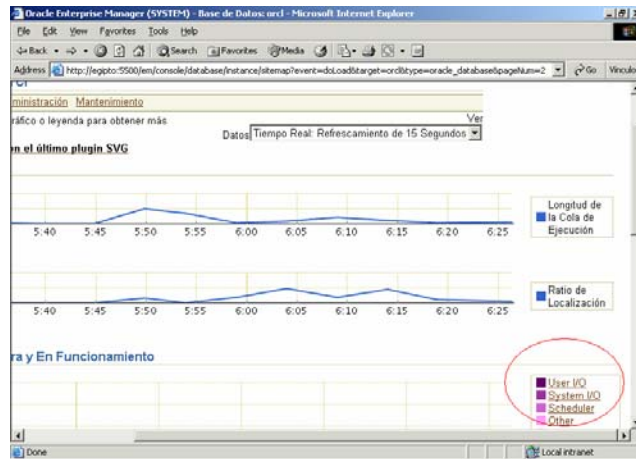
| | |
|---|--|
| | <p>cantidad de blocks consumidos y el total de registros. Y en caso de los índices se actualizan el LEAF_BLOCKS, BLEVEL y otros.</p> <p>DBA_THRESHOLDS = Muestra las métricas con los warnings y valores críticos.</p> |
| <p>Establece la velocidad promedio de lectura de un simple bloque database. Esto permite indicarle al ADDM la velocidad de lectura el cual lo tomará para sus calculos y recomendaciones.</p> | <p>EXEC DBMS_ADVISOR.SET_DEFAULT_TASK_PARAMETER('ADDM','DBIO_EXPECTED',#); El número esta expresado en milisegundos. El default es 7000 milisegundos.</p> |
| <p>Activar y desactivar la visualización del plan de ejecución para los queries.</p> | <p>set autotrace traceonly; set autotrace off;</p> |
| <p>Setea la pantalla del SQL Plus para los reportes.</p> | <p>SET SERVEROUTPUT ON SIZE 999999 SET LONG 999999</p> |

RECOMENDACIONES

- No se debe usar memoria SWAP en el SGA.
- No se debería ir más allá del 60% de la memoria RAM.
- Al ingresar al Database Control hay un link llamado Resumen de Diagnóstico donde este tiene un link llamado Busquedas de Rendimiento (Performance Findings), ahí nosotros podemos dar click y nos motrará los problemas de performance encontrados para afinar.

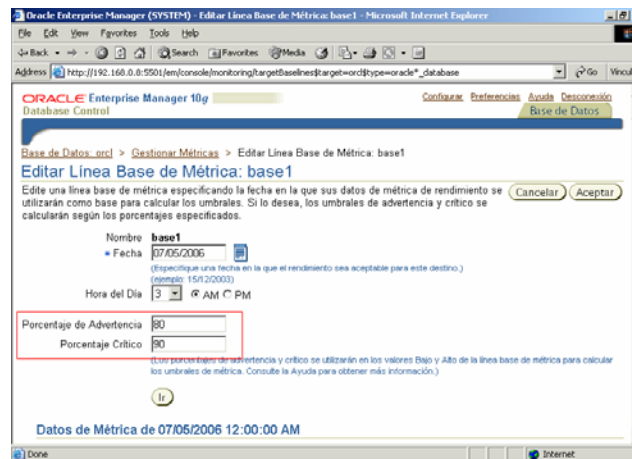


- En el Database Control hay un tab llamado Rendimiento donde se encuentra una tabla con todos los aspectos que se pueden analizar información importante.



- Para realizar un reporte personalizado de un snapshot con respecto a los n anteriores ingresar al Database Control luego al tab Rendimiento, luego en la opción Central de Asesores (Parte inferior) y luego en el link ADM. Luego seleccionar una hora de inicio y una hora final para el análisis. Para generar el reporte hay que dar clic en el botón Ver Informe. En la pantalla si se toma como base 2 snapshots mostrará la misma información localizada al iniciar el Database Control llamado Resumen de Diagnóstico.

- Creando un base line. Al ingresar el porcentaje cuando sea un warning o crítico automáticamente lo considera para todas las métricas de ese baseline. Al ingresar un porcentaje se lanzara el warning o el critical cuando la cantidad llegue al valor de la metrica * 1.n, donde n es el porcentaje de la métrica.



- Debemos ubicar los objetos que están en estado INVALID ó UNUSABLE en la base de datos para corregirlos.
 - select owner, object_name, object_type from dba_objects where status = 'INVALID' or 'UNUSABLE';
 - alter view ó procedure ó trigger etc nombre_objeto compile;
 - alter index nombre rebuild; (Nota: Al hacer rebuild, el índice crece en un 50% su tamaño temporalmente).

BACKUPS

CONCEPTOS

- MTTR = Tiempo para un recovery, MTBF = El tiempo de diferencia entre una falla y otra.
- Con Real Application Cluster (RAC) y Oracle Streams (Replicación) conseguimos mayor escalabilidad.
- Debemos disminuir el MTTR (Tiempo de recovery).
- Eliminamos la pérdida de transacciones comiteadas utilizando archived redo logs, standby databases (registra los archivers redologs) y Oracle Data Guard.
- Los control files son archivos binarios que contienen la estructura de la base de datos. El control file es actualizado cuando hay cambios físicos, ejemplo: renombrar un datafile. Los control files almacenan el nombre de la base de datos y su creación, la localización de los datafiles, redo logs, lista de los tablespaces, el log secuencial (es incrementado cuando hay un switch de log files), el más reciente checkpoint, el inicio y final de los undo segments y información del último backup.
- Cuando no está en modo archiver el Log Writer Process es el que escribe en el control file cual es el actual log file y cuando se está en modo archiver el CKPT Process es el responsable de hacer esto.
- El control file tiene 2 secciones llamadas reusable y no reusable, cuando se utiliza RMAN sin catálogo la información se almacena en la sección reusable.
- El CKPT process controla la cantidad requerida para un instance recovery.
- Un checkpoint, el CKPT actualiza el control file y las cabeceras de los datafiles para que reflejen el mismo SCN. Todos los dirty buffer blocks son bajados a disco por el DBWriter. Ocurre esto cuando un tablespace es puesto a readonly, offline ó backup. También ocurre cuando ocurre un switch entre los log files.
- Los redo logs registran los cambios de la base de datos y en algunos casos antes de ser escrita en los datafiles. Los redo logs son llenados con redo records.
- Los procesos Redo Log Writer escriben la información del redo log buffer a los online redo logs por las siguientes circunstancias: cuando se hace un commit, cuando el log buffer está 1/3 de lleno y cuando el buffer alcanzó 1 MB de registros actualizados (no inserts ni deletes) solo updates. El Redo Log Writer escribe en los redo log files antes que el DBWriter escriba del Buffer Cache a los datafiles.
- La caída del LGWR o el ARC# produce una caída de la base de datos pero gracias a los redologs no perdemos las transacciones comiteadas.
- Cada instancia en un ambiente RAC tiene 1 online redo thread.
- Cuando se genera un switch entre los logfiles; Oracle escribe en el alert de la base de datos y genera un nuevo log secuencial y lo asigna a un grupo de logfile.
- Un v\$log con el status current indica que es el actual logfile que se está escribiendo. Si v\$logfile con el status active indica que el logfile es requerido para instance recovery, si está invalid significa que está perdido, si es deleted significa que el redo log ya no será usado, stale significa que el miembro del grupo es nuevo y no ha sido usado y el status blank indica que el file está en uso.
- Si falla un miembro de los logfiles no afecta a la base de datos. Al menos debe tener 1 miembro cada grupo para que la base de datos no falle.
- Cuando se agrega un miembro a un grupo no se especifica el tamaño porque este adopta el tamaño de los miembros del grupo.

- Si eliminamos un grupo logfile o un miembro, el grupo debe tener el status de INACTIVO. Si solo hay 2 grupos no deberiamos borrar el último miembro del grupo.
- Si uno de los procesos de archiver falla, la instancia tambien fallará.
- Con RMAN podemos establecer políticas para definir el numero de días para que un backup sea obsoleto (window policy) o redundancy policy (múltiples copias de cada backup) en pro de no tener problemas de el espacio que usará el archivers.
- Para poner una base de datos en modo archiver debemos estar en modo MOUNT.
- El Flash Recovery Area es un area que unifica todos los archivos relacionados a actividades de recovery del Oracle. Puede ser compartido por varias bases de datos. Ahí se almacena la primera vez que se crea la base de datos una copia del control file. Además el parametro ARCHIVE_DEST_10 apunta a la carpeta del Flash Recovery Area y tambien los archivos relacionados a la tecnologia Flash Back. Si nos quedamos sin espacio para almacenar más archiverlogs la base de datos estara indisponible.
- Cuando en el RMAN se habilita la opción Control File Autobackup, esto permite que cualquier backup a la base de datos o parte de ella el control file es incluido dentro del backup.
- En un ambiente RAC todas las instancias deben tener el mismo valor del DB_RECOVERY_FILE_DEST_SIZE y el DB_RECOVERY_FILE_DEST.
- Cuando archivos son borrados del Flash Recovery Area por ser obsoletos es registrado en el alert. Si no hay archivos por eliminar y el Flash Recovery alcanza un 85% o un 97% es colocado en el alert tambien.
- Los tipos de backups son: **whole database** (Todos los datafiles y un control file, online redo log nunca son copiados) y **partial** (Cero o más datafiles). Además **full** (Incluye todos los bloques de cada datafile en un whole o partial base de datos) o **incremental** (Solo copia los bloques cambiados desde un ultimo backup). Tambien pueden ser **consistente** (Conocido como un offline backup, es llevado mientras la base de datos no esta open, es consistente porque el SCN es el mismo en el control file y en las cabeceras de los datafiles) e **inconsistentes** (Conocido como online backup, es ejecutado mientras la base de datos esta en open y disponible para los usuarios, es inconsistente porque el SCN no esta sincronizado).
- Oracle tiene 4 niveles de backups incrementales. El nivel 0 es considerado un base line o full backup.
- Cuando se genera un backup del controlfile se guarda en el USER_DUMP_DEST.
- Los tipos de disk backup son: image copy, backup set y compressed backup set. Los backup set hacen backup a los bloques utilizados unicamente y puede ser redireccionado a un tape backup.
- Un Incrementally Updated Backup es crear un image backup de los datafiles y luego progresivamente actualizar las copias de los datafiles con backups incrementales. Esto permite utilizar menos redos para un recovery.
- Solo se puede hacer backups incrementales a un tablespace, datafiles y a una base de datos completa.
- Un backup hecho en NOARCHIVE no puede ser usado en modo ARCHIVE para recover.
- Si estoy en modo NOARCHIVE y los redo no han hecho un ciclo de switch, puedo restaurar hasta la ultima transacción a pesar que se esta en modo NOARCHIVE.

- En el EM Database Control en la sección Estrategia Shedule Backup solo podemos hacer backup a los siguientes objetos: datafiles, archive logs, tablespaces y la base de datos completa.
- En modo NOARCHIVE solo se puede hacer backups completos.
- Si hacemos un backup por el sistema operativo podemos usar el comando CATALOG de RMAN para ingresarlo al repositorio del mismo.

PARAMETROS

| Parametro | Objetivo |
|---|---|
| CONTROL_FILE_RECORD_KEEP_TIME = # | Indica el tiempo que se almacena una información en la sección reusable del control file. El default es 7 días. |
| MAXLOGFILES = # | Indica la cantidad de logfiles como maximo que se puedan asignar a la base de datos. |
| MAXLOGMEMBERS = # | Especifica la cantidad maxima de miembros en un grupo. |
| FAST_START_MTTR_TARGET = # | Asegura que el tiempo de un instance recovery no exceda. |
| LOG_ARCHIVE_MIN_SUCCEED_DEST =# | Indica cuantos destinos deben grabarse los archiver logs satisfactoriamente. |
| LOG_ARCHIVE_FORMAT = “%d(id de la ase de datos)_%s(número de log secuencia)_%t_%r(asegura un número único)” | Especifica el formato de los archive logs para asegurarse que no sean sobre escritos. |

SCRIPTS

| Objetivo | Script |
|---|---|
| Modifica la ubicación y numeros de control files de la base de datos. | ALTER SYSTEM SET CONTROL_FILES='ruta1','rutan' SCOPE=SPFILE; |
| Genera un switch. | ALTER SYSTEM SWITCH LOGFILE; |
| Ordena un checkpoint. | ALTER SYSTEM CHECKPOINT; |
| Muestra los grupos activos de los redo log. | v\$logfile; |
| Crea un nuevo grupo de logfile. | ALTER DATABASE ADD LOGFILE GROUP # ('ruta1','rutan') SIZE # ó REUSE; |
| Agregando un miembro a un grupo. | ALTER DATABASE ADD LOGFILE MEMBER 'ruta' TO GROUP #; |
| Eliminando un grupo. | ALTER DATABASE DROP LOGFILE GROUP #; Luego eliminar mediante el S.O. |
| Eliminando un miembro. | ALTER DATABASE DROP LOGFILE MEMBER 'ruta'; Luego eliminar mediante el S.O. Si estamos en archiverlog antes de borrar el miembro el grupo debe ser guardado. |
| Limpia los online redo logs. | ALTER DATABASE CLEAR LOGFILE GROUP #; ALTER DATABASE CLEAR UNARCHIVED LOGFILE GROUP #; (Antes de usar el UNARCHIVED es mejor hacer backup completo). |
| Guarda un grupo. | ALTER SYSTEM ARCHIVE LOG GROUP #; |
| Agregando un destino a los archiver log. | ALTER SYSTEM SET LOG_ARCHIVE_DEST_n="(LOCATION='ruta') MANDATORY REOPEN"; La opción mandatory indica que la grabación en |

| | |
|---|---|
| | ese destino se debe realizar. La opción REOPEN que si no se realizó la anterior vez, esta vez debe realizarse. |
| Visualizar si la base de datos esta en modo archiver. | v\$database (log_mode) |
| Poner la base de datos en modo archiver. | ALTER DATABASE NOARCHIVELOG; (En estado mount). |
| Especifica la ruta del Flash Recovery Area. | ALTER SYSTEM SET DB_RECOVERY_FILE_DEST = 'ruta'; |
| Especifica el tamaño del Flash Recovery Area. | ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE = #; |
| Muestra los warnings publicados en el alert por el Flash Recovery Area. | SELECT * FROM DBA_OUTSTANDING_ALERTS; |
| Backup del controlfile y lo guarda en el USER_DUMP_DEST. Consiguiendo el script del control file podemos restaurar el control file en caso de perderlo ó podemos usarlo para cambiarle el nombre a la base de datos. | ALTER DATABASE BACKUP CONTROLFILE TO TRACE; |

RMAN

| Objetivo | Comando |
|--|--|
| Ingresando al RMAN. | rman target / |
| Muestra todas las configuraciones disponibles. | RMAN> show all; |
| Muestra información de los archive logs. | SQL> archive log list; |
| Lista los backups realizados. | RMAN> list backup; ó RMAN> list backup of database ó datafile 'ruta'. |
| Lista todos los archive | RMAN> list archivelog all; |
| Lista un resumen. | RMAN> list backup summary; |
| Lista los archivos que necesitan hacerle backup. | RMAN> report need backup; ó RMAN> report need backup level ó 1. |
| Lista los backups obsoletos. | RMAN> report obsolete; |
| Permite generar backups incrementales level 0 ó 1. | RMAN> backup incremental level 0 ó 1 as compressed backupset database; |
| Elimina todos los backups que no son necesarios. | RMAN> delete noprompt obsolete; |
| Información de los backups. | V\$recovery_file_dest. (Campos: name, space_limit, space_used, space_reclaimable y number_of_files). |
| Realiza un backup incremental. | RMAN> backup incremental level # tablespace nombre datafile 'ruta1, ruta2, rutan' ó database, etc. |

IMAGENES



RECOMENDACIONES

- Los logfiles deberían tener el mismo tamaño como recomendación.
- Tener cada miembro de los redolog en disco separados, si en caso tenemos un redolog y un datafile en el mismo disco no debemos poner el UNDOTBS, SYSAUX, SYSTEM y algún tablespace asignado para índices en el mismo disco.

RECOVERY

CONCEPTOS

- Media failure es cuando uno o mas archivos de la base de datos son perdidos o llegan a estar corruptos.
 - Instance failure es cuando la instancia de la base hace shutdown y los archivos de la base de datos no estan sincronizados. Esto requiere un recovery la siguiente vez que se levante la instancia. Esto ocurre cuando un process background cae, un shutdown abort o caida del sistema operativo.
 - Un objeto eliminado se ubica en el recycle bin y es almacenado en una sección del tablespace hasta que el tablespace necesite de ese espacio para otros objetos.
 - Cuando deseamos entrar a la fase NOMOUNT se lee el archivo pfile o spfile, en la fase MOUNT se busca los controlfiles y deben estar todos los controlfiles multiplexados y OPEN donde todos los datafiles cn estado online deben estar disponibles y al menos un miembro de cada grupo de los redo log debe estar disponible.
 - Los datafiles pertenecientes a los tablespaces SYSTEM y UNDO no puede ser puestos en offline.
 - Si Oracle no puedes sincronizar el SCN utilizando los redologs (instance recovery) pasa a necesitar un media recovery utilizando los archivers redo logs.
 - **La instancia cae cuando uno de los control files se pierde, un grupo de redo logs esta perdido y cuando uno de los datafiles del tablespace SYSTEM y del UNDO es perdido.**
 - STARTUP FORCE = SHUTDOWN ABORT + STARTUP.
 - El instance recovery es ejecutado en la fase OPEN y previamente ya se levantó las transacciones commit y uncommit de los redo log files. Despues del OPEN se usa el undo para rollback las transacciones uncommit.
 - El recycle bin es un estructura logica dentro de cada tablespace que lleva las tablas borradas y los objetos relacionados a la tabla como los índices. Cuando ya no hay espacio disponible para el tablespace para ingresar nuevos objetos, Oracle comienza a eliminar (Modo FIFO) los objetos del recycle bin. Las restricciones que tiene son las siguientes:
 - Los SYSTEM locally managed tablespace no pueden tener recycle bin, por lo tanto no se envian objetos que se ubiquen en el tablespace system al recycle bin cuando se eliminan.
 - Los objetos dependientes de una tabla son llevados al recycle bin si se elimina primero la tabla. Por ejemplo: Si primero se elimina un índice este no es llevado al recycle bin. No se cumple para los bitmap indices, referencia de integridad referencial y log de las vistas materializadas.
- El recycle bin se utiliza para hacer los flashback drop.
- El FlashBack table utiliza los segmentos undo para hacer rollback a la tabla hasta un tiempo determinado.
 - Los datafiles críticos en la base de datos son SYSTEM y UNDO y provoca la caida de la base de datos.
 - En modo noarchivers se tiene que restaurar toda la base de datos y no una parte de ella.
 - Para recuperar un datafile crítico debemos bajar a MOUNT si estamos en modo archiver.
 - Cuando se llena la carpeta del flash_recovery_area la base de datos se detendrá. En el flash_recovery_area esta los archive logs, los control files y los backups de la base de datos.

- En un proceso de recovery, para el Oracle es más fácil hacer recovery a un image copy que un backup set, porque hay un proceso de extracción de los datafiles en el dataset.

PARAMETROS

| Parametro | Objetivo |
|----------------------------|--|
| FAST_START_MTTR_TARGET = # | Indica el tiempo que se demora para hacer un instance recovery o un check point. |

SCRIPTS

| Objetivo | Script |
|---|---|
| Muestra todos los datafiles que no se pueden leer al querer levantar la base de datos en estado open. | v\$recovery. (El campo file# nos da el número de datafile que se refiere, asociarlo con la vista v\$datafile). |
| Tamaño recomendable del redo log. | v\$instance_recovery (El campo OPTIMAL_LOGFILE_SIZE indica el tamaño recomendable en megas del redo log, el campo ESTIMATED_MTTR indica el tiempo para recuperar una instancia en ese momento). |
| Consultando una tabla en el tiempo. | select * from tabla as of timestamp (valor en timestamp). |
| Indica la cantidad de espacio libre en el tablespace. | dba_free_space. |
| Consultamos el recycle bin. | select * from recycle_bin ó dba_recyclebin; |
| Limpia el recyclebin. | purge recyclebin; |
| Limpia un objeto del recyclebin. | purge table "BIN\$xxxxx"; |
| Permite recuperar una tabla que se encuentra en el recycle bin. | flashback table nombre ó "BIN\$/xxxxx" to before drop rename to nombre_alternativo; |
| Permite recuperar uno o varias tablas que se encuentra en el undo. | alter table tabla enable row movement; flashback table tabla1, tablan to timestamp (valor en timestamp) ó to scn #; Ejemplo: flashback table tabla to timestamp '2006-05-12 10:00:00'; |
| Devuelve información (Ejemplo: Espacio que ocupa, etc) sobre los tipo de datos. | v\$type_size |
| Devuelve el actual SCN de la base de datos. | v\$database (Campo current_scn). |
| Devuelve el valor del id de la transacción generado al hacer una operación DML. | select versions_xid, versions_operation from tabla versions between SCN minvalue and maxvalue; |
| Devuelve la sentencia SQL que provoca deshacer la sentencia DML ingresada por el id de una transacción. | select operation, undo_sql, logon_user from flashback_transaction_query where xid = HEXTORAW('valor_row_id_transaccion'); |
| Asociación de un índice con una tabla. | dba_ind_columns; |
| Muestra los valores del spfile. | v\$spparameter (campos: name, value). |
| Devuelve la asociación de los tablespaces con sus datafiles. | select t.name, d.name from v\$tablespace t join v\$datafile d using (ts#); |
| Indica el estado de los archive process. | v\$archive_processes. |
| Hace un media recovery a un datafile. | alter database recover datafile 'ruta'; |

RMAN

| Objetivo | Comando |
|-----------------------|---|
| Restaura un datafile. | <pre>RMAN> run {sql 'alter database datafile # offline'; restore datafile #,..n; recover datafile #,..n; sql 'alter database datafile # online'; }</pre> |

LOG MINER

- Setear el parametro utl_file_dir a un directorio. (ALTER SYSTEM SET UTL_FILE_DIR = 'carpeta' SCOPE=SPFILE;
- Copiar los redo logs a examinar a la carpeta del utl_file_dir.
- EXEC SYS.DBMS_LOGMNR.D.BUILD('dictionary.ora','carpeta');
- EXEC
SYS.DBMS_LOGMNR.ADD_LOGFILE('ruta_logfile1',SYS.DBMS_LOGMNR.NEW);
- Si hubiera mas redo logs:
EXEC
SYS.DBMS_LOGMNR.ADD_LOGFILE('ruta_logfile1',SYS.DBMS_LOGMNR.ADDFILE);
- Para comenzar el analisis:
EXEC
SYS.DBMS_LOGMNR.START_LOGMNR(DICTFILENAME=>'ruta_utl_file_dir/dictionary.ora');
- SELECT * FROM v\$LOGMNR_CONTENTS; Campos importantes:
TIMESTAMP, SEG_OWNER, SEG_NAME, OPERATION, SQL_REDO,
SQL_UNDO.

RECOMENDACIONES

- No debemos tener mas del 90% del tamaño del redo log más pequeño sin hacer un checkpoint.
- Oracle recomienda tener los siguientes parámetros deshabilitados:
 - LOG_CHECKPOINT_TIMEOUT, indica la cantidad de segundos que un nuevo bloque modificado o nuevo en el buffer cache espera para ser escrito en disco.
 - LOG_CHECKPOINT_INTERVAL.
 - FAST_START_IO_TARGET, indica lo mismo que el FAST_START_MTTR pero el número indica el número de I/O en lugar de segundos para finalizar un instance recovery.
- Para recuperar un miembro redo log.
 - Visualizar cual es el miembro perdido: select * from v\$logfile;
 - alter system archive log group #;
 - alter database clear logfile group #;