Configuración Oracle

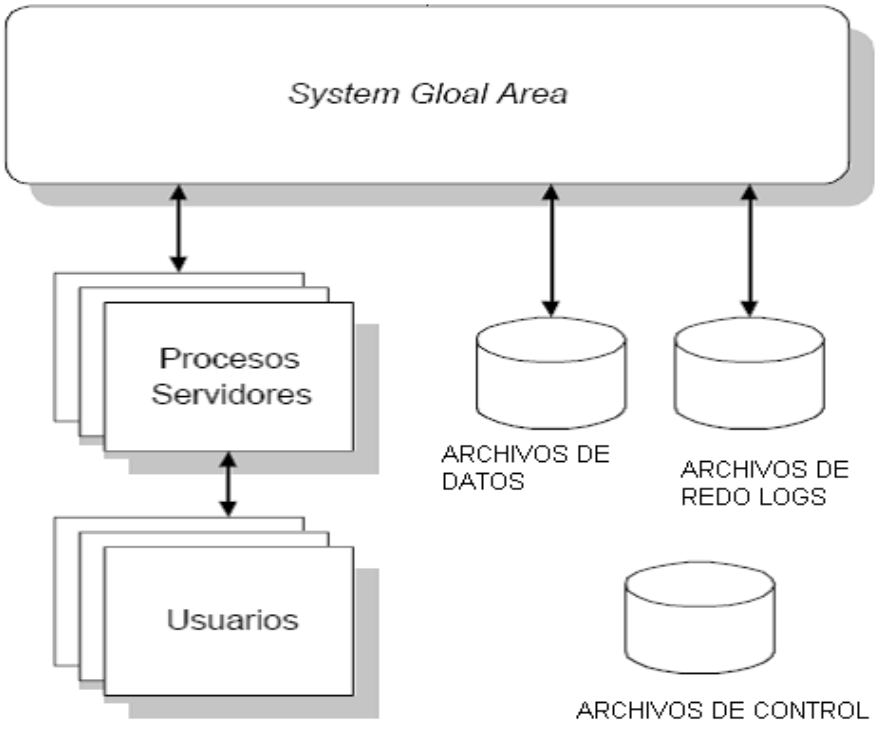
1.- INTRODUCCIÓN

Una BBDD, físicamente y en su forma mas simple, no es mas que un conjunto de archivos ubicados en uno o mas discos físicos, gestionados por una serie de procesos. La ubicación y distribución física de estos ficheros es irrelevante para la BBDD, pero importante para el rendimiento de esta. El formato de los ficheros que componen la BBDD, es binario, lo que significa que solamente pueden ser accedidos por el núcleo de la base de datos.

Un Servidor Oracle tiene dos elementos distintos, la **instancia** y la **base de datos.**

- Base de datos: conjunto de archivos físicos para el manejo de los datos.
- Instancia: Estructura de memoria y conjuntos de procesos.

Además de los dos elementos anteriores oracle proprociciona un ENTORNO CLIENTE/SERVIDOR, el cual es un sistema basado en compartir aplicaciones y/o datos a través de una red.



2.- BASE DE DATOS:

Una base de datos oracle está formada por:

- Uno o varios archivos de datos
- Al menos un archivo de control
- Dos grupos o mas de archivos trazas o redologs

2.1 ARCHIVOS DE DATOS (DATAFILE)

Son aquellos que almacenan los datos reales, es decir, los objetos del usuario. También almacenan el diccionario de datos. Normalmente son los de mayor tamaño.

Una base de datos ORACLE debe poseer al menos un archivo de datos.

Un datafile puede tener cualquier nombre y extensión (siempre dentro de las limitaciones del sistema operativo) y puede estar localizado en cualquier directorio del disco duro

2.2 ARCHIVOS DE CONTROL

Es un archivo binario de tamaño reducido y estable, que contienen la información necesaria para mantener y verificar la integridad de la base de datos, como la ubicación de los datos y los Redo Log.

ORACLE requiere de esta información cada vez que se “arranque” la instancia. Toda base de datos ORACLE requiere por lo menos un Control File.

Oracle, por defecto, crea 2 ficheros de control que son iguales (es decir, están multiplexados). Si los ficheros de control multiplexados no son iguales, oracle dará un error de sincronización.

La información que contiene es la siguiente:

- Nombre de la base de datos
- Fecha de creación y nombre de la base de dato.
- Información de arranque y parada
- Localización y nombre de los archivos de la Base de datos y Redo log
- Información de los puntos de control (checkpoints). Los checkpoints se da cuando se llena los Redo log, se detiene la base de datos, etc.
- Estado de los online y offline de los archivos.

Es esencial que los Control Files estén protegidos y que haya múltiples copias en distintos discos (mínimo dos) ya que en caso de perdida no se podrá reiniciar la instancia de la base de datos con la consecuente pérdida de toda la información.

2.3.- ARCHIVOS TRAZAS O REDO LOG

Mantienen información de todos los cambios efectuados sobre la base de datos para asegurar su reconstrucción en caso de fallo. Tiene que haber por lo menos dos grupos Redo Log Files y uno de ellos debe estar activo, online. Se escribe en ellos de manera circular.

Es esencial que los Redo Log posee un buen rendimiento y estén protegidos contra fallos de hardware. Si se pierde la información contenida en estos archivos será imposible la recuperación de la base de datos en caso de fallo del sistema.

En caso de fallo sin perdida de datos, como cuando falla la máquina, ORACLE puede aplicar la información de los registros de rehacer en línea (redolog on-line) automáticamente sin intervención del administrador de la base de datos.

Los registros de rehacer en línea están siempre en uso mientras la instancia de la base de datos está activada. Los cambios que se realizan en la BBDD, son registrados por turnos, cuando uno esta lleno, se escribe en el otro, y cuando este otro esta lleno se sobrescribe en el primero.

2.3.1.-Multiplexado de los redo log

Los correspondientes redolog on-line son llamados grupos. Cada fichero redolog on-line dentro del grupo es llamado miembro y los miembros de un grupo son del mismo tamaño.

Es decir, cuando utilizamos la forma multiplexada, lo que estamos haciendo es escribir la misma información en cada miembro de un grupo, protegiéndose así contra fallos en el disco.

2.3.2.-Archivados de los redo log

Oracle permite poder salvar los grupos de redolog on-line hacia uno o más destinos en off-line, estos ficheros se conocen como redolog archivados o logs archivados.

Los archivos de redolog on–line se archivan una vez que se llena. Cuando estamos en este modo, llamado modo ARCHIVELOG, no se puede escribir en el siguiente grupo de redolog on-line si éstos aún no han sido archivados.

El modo archivado nos permite:

- Recuperar la base de dato en el tiempo (aplicando los logs archivados)
- Realizar copias de seguridad en caliente
- Conseguir información sobre la base de datos.

2.4.- OTROS ARCHIVOS FÍSICOS IMPORTANTES

Estos archivos son necesarios, pero son externos a la base de datos

SPFILE. (Archivos de inicialización del servidor)

Sus entradas son específicas para la instancia que está siendo accesada. Sin este archivo la instancia de oracle, no puede arrancar. Es muy pequeño pero debe existir

LOG.XML.

Contiene sucesos de la gestión de la base de datos cronológicamente y mensajes críticos que se producen en la instancia y base de datos. Oracle sigue manteniendo el archivo alert.log de versiones anteriores por compatibilidades

TRACE FILES

Almacena las trazas de los procesos de la instancia.

PASSWORD FILE

Lo normal es que los usuario y passwords se almacenen en el diccionario de datos, pero cuando no está disponible (en el inicio del arranque) se usa este fichero para leerlos

LISTENER.ORA

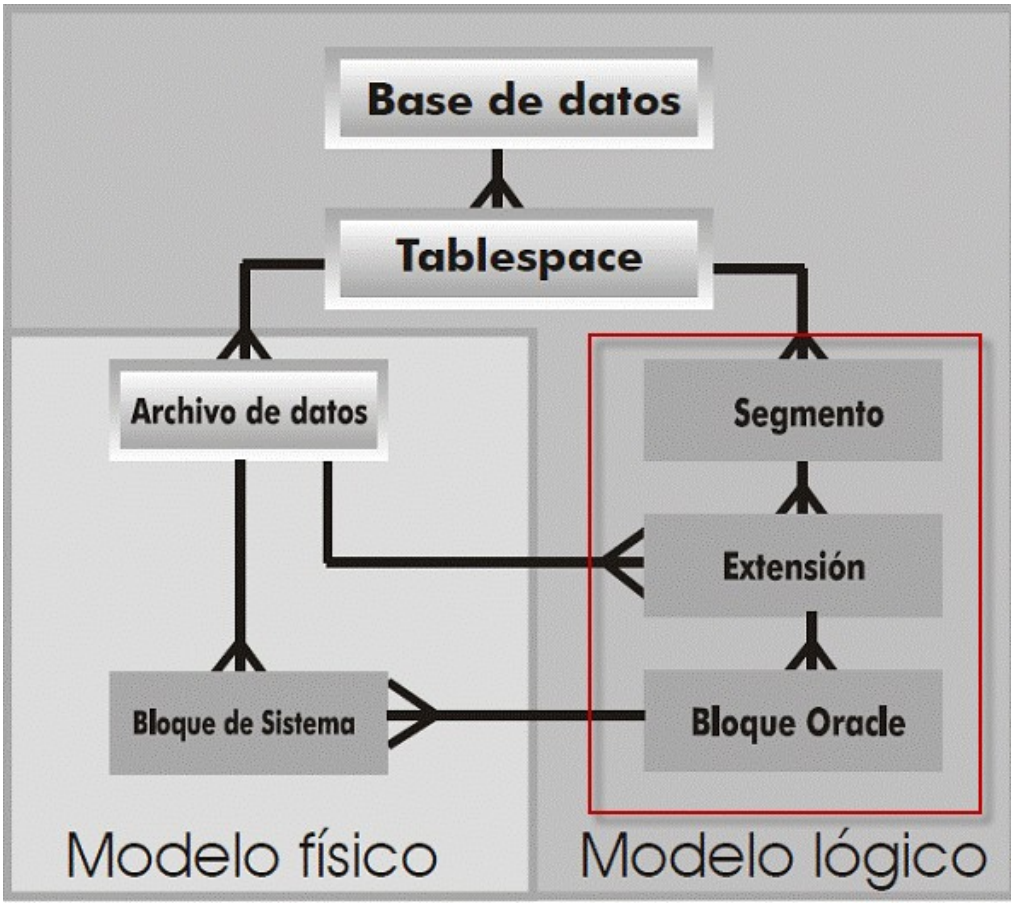
Archivo de escucha del servidor, que provee la conectividad de red con la base de datos *Oracle*.

TNSNAMES.ORA

Archivo de petición del cliente. Se comunica con el listener para poder acceder al servidor oracle

3.- ESTRUCTURA LÓGICA DE ALMACENAMIENTO

A nivel lógico el elemento mínimo lo define el bloque de Oracle, el cual a su vez puede estar formado por varios bloques del Sistema Operativo. Con varios bloques de Oracle se forman las extensiones las cuales darán lugar a los segmentos (objetos de la base de datos: tablas, índices, etc.); los segmentos están contenidos en los tablespaces los cuales forman la base de datos.



a.- Bloques de datos.

Un bloque es la unidad mas pequeña de entrada/salida de almacenamiento usada por Oracle. Cada bloque está formado por un número determinado de bloque del sistema operativo.

A la hora de crear una base de datos se debe indicar cuántos bloques del sistema operativo formará un bloque de datos. Es muy importante decidir este valor de antemano ya que una vez creada la base de datos ya no se puede modificar excepto en migraciones mas actuales del producto.

Oracle recomienda que el tamaño de un bloque de datos sea siempre un múltiplo del bloque de datos del sistema operativo.

Un bloque de datos es la mínima unidad de Lectura/Escritura en una base de datos Oracle, es decir, Oracle no lee y escribe en bloques del sistema operativo sino en los bloques de Oracle, los cuales puede variar de una base de datos a otra en la misma máquina.

b.- Extensiones y Segmentos

Las extensiones es un unidad lógica de almacenamiento que está formada por un número determinado de bloques de datos contiguo.

La agrupación de una o varias extensiones (no tienen que ser contiguas) forman un segmento que pueden ser de diferentes tipos:

- Segmentos de datos: Cada segmento de datos almacena los datos correspondientes a una tabla.
- Segmentos de índice: Cada segmento de índice mantiene los datos para un índice definido dentro de la base de datos.
- Segmento de Rollback: son segmentos de Rollback permite almacenar las acciones de una transacción que deben ser deshechas bajo ciertas circunstancias.
- Segmentos Temporales: Los segmentos temporales se crean cuando se requiere de un espacio temporal para procesar una instrucción de SQL, y son destruidos una vez que haya culminado el procesamiento de la instrucción. Oracle sólo utilizará este segmento cuando no pueda realizar la consulta en memoria o cuando no pueda buscarse un método alternativo para realizarla, utilizando índices. Las consultas en la que Oracle se ve en la obligación de utilizar los segmentos temporales son: select……order by, select….group by, create index, union, minus, intersect, distinct.

Nota: los datos de una tabla están en un solo segmento de tipo datos, que a su vez estará formada por una o varias extensiones y cada una de estas extensiones está formada por un número determinado de bloques.

b.1.- Funcionamiento

Cuando se crea un segmento por primera vez, se crea obligatoriamente una extensión en dicho segmento. Pero a medida que va creciendo el segmento se va asignando nuevas extensiones al mismo. Al asignar una nueva extensión, se está reservando espacio en disco para almacenar los nuevos datos de dicho segmento. Por tanto, al crear la nueva extensión, está totalmente vacía y todo su espacio está disponible para almacenar los datos del segmento y, además, en el disco debe haber espacio libre para que Oracle reserve todo el tamaño que necesite la extensión y la formatee de forma especial para poder utilizarla. A partir de ese momento, en esa extensión sólo se podrán almacenar datos del segmento al que pertenece.

Cuando se llenan todos los bloques de datos de una extensión, el segmento solicita una nueva extensión al sistema para poder seguir almacenando la información.

La razón principal de esta estructura es la de minimizar el espacio desperdiciado (vacío) de un tablespace. A medida que se insertan o eliminan filas de una tabla, las extensiones del tablespace asociado a la tabla pueden aumentar o disminuir de tamaño. De esta forma el espacio para el almacenamiento de los datos puede ser administrado dependiendo de cómo la tabla va sufriendo modificaciones en el número de filas.

Cuando se crea un tablespace se puede especificar el número mínimo de extensiones a ser asignados a sus objetos, así como el número de extensiones a agregar cada vez que se agote el espacio disponible para almacenar datos.

c.- Tablespaces

Un tablespace (TB) o espacio de tabla es una unidad lógica de almacenamiento dentro de una base de datos oracle (podríamos decir que es una asociación entre el sistema de ficheros del sistema operativo y la base de datos).

No es un fichero físico en disco.

Un TB sólo puede pertenecer a una base de datos y puede contener 1 o mas ficheros de datos (Datafile). Al utilizar más de un Datafile por tablespace puede distribuirse los datos sobre varios discos y balancear la carga de E/S, mejorando así el rendimiento del sistema.

Un Datafile sólo puede pertenecer a un tablespaces.

Los espacios de tabla se utilizan para mantener juntos los datos de usuario o de aplicaciones, para facilitar su mantenimiento o mejorar las prestaciones del sistema. Por ejemplo, puede tenerse un tablespace para almacenar los datos de contabilidad y otro para almacenar los datos de ventas en una empresa de servicios. Al segmentar grupos de datos relacionados en tablespaces diferentes se simplifican las tareas de administración de dichos grupos. Se aconseja que cada aplicación tenga su propio espacio de tabla

El Administrador los usa para:

- Controlar la creación de espacios en disco para los datos de la BD.
- Asignar cuotas específicas para los usuarios.
- Controlar la accesibilidad de los datos (poniendo un tablespace en modo online/offline o read-only/read-write). Pueden estar montados sobre dispositivos ópticos si son de lecturas.
- Realizar operaciones parciales de backups/restore.
- Repartir los datos en varios discos para mejorar el rendimiento.

Cuando se crean se les asigna un espacio en disco que Oracle reserva inmediatamente, se utilice o no. Si este espacio inicial se ha quedado pequeño, Oracle puede gestionar el crecimiento dinámico de los ficheros sobre los que se asienta los espacios de tabla.

Como parte del proceso de crear la base de datos, ORACLE automáticamente crea, entre otros, los siguientes tablespaces:

SYSTEM.- Almacena todos los datos del sistema, el catálogo. No puede ponerse en off-line
Siempre está on-line cuando la base de datos está abierta

SYSAUX.- Es un espacio de tabla auxiliar del tablespace SYSTEM.. Algunos componentes que utilizaba el tablespace SYSTEM en versiones anteriores a ORACLE 10g, están almacenados aquí. Todas las bases de datos 1Og o versiones posteriores deben tener un espacio de tabla SYSAUX.

UNDO o ROLLBACK.- Almacena los segmentos rollback (deshacer)
No permite almacenar ningún otro tipo de segmentos (tablas, indices)

TEMP.- Almacena las tablas temporales

USER.- Contiene información personal de los usuarios

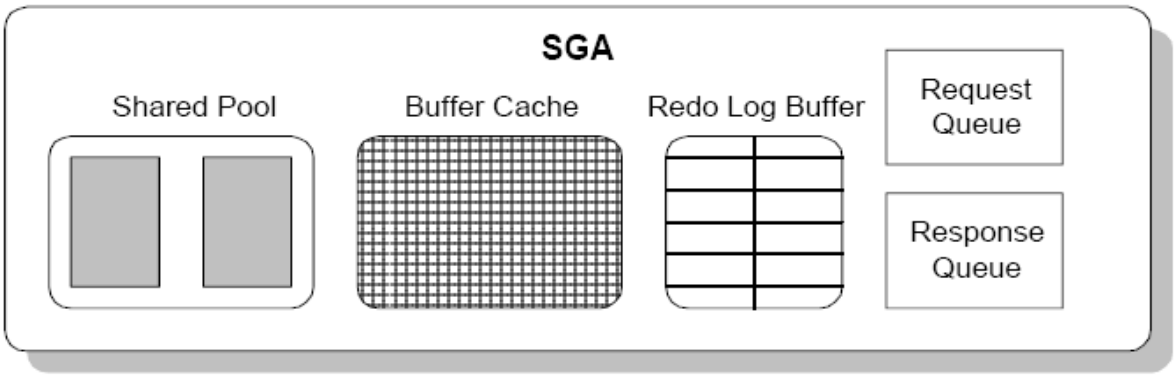
4.- ESTRUCTURA LÓGICA DE MEMORIA

Una Instancia está formada por:

- SGA (System Global Area). Es una zona de memoria compartida.
- BACKGROUND PROCESS. Son un conjunto de procesos en segundo plano.
- SERVER PROCESS. Conjunto de procesos servidores, los cuales se encargan de procesar las consultas de los usuarios.

4.1.- Estructura de la SGA

A modo General, se compone de varias zonas:



a) Bufer bloque de datos (DataBase Buffer Cache)

Su función es mantener bloques de datos leídos directamente de los archivos de datos. Cuando se procesa una consulta, el servidor busca los bloques de datos requeridos en esta estructura. Si el bloque no se encuentra en esta estructura, el proceso servidor lee el bloque de la memoria secundaria y coloca una copia en esta estructura. De esta forma, otras peticiones que requieran de este bloque de datos no requerirán de acceso a memoria secundaria, con lo que ser realizarán menos lecturas físicas.

El Database Buffer Cache está organizado en dos listas:

1. Lista de sucios.- mantiene aquellos bloques de datos que han sido modificados y que aún no han sido escritos a disco.
2. Lista de menos recientemente usados.- mantiene los bloques libres, los bloques a los que se está accediendo actualmente y los bloques sucios que aún no han sido remitidos a la lista de sucios.

b) Bufer de registro deshacer (Redo Log Buffer)

Es un buffer circular que mantiene todos los cambios que han sido realizados sobre la base de datos por operaciones de insert, update, delete, create, alter y drop. Las entradas de este buffer contienen toda la información necesaria para reconstruir los cambios realizados a la base de datos por medio de cualquier sentencia del DDL o del DML (el bloque que ha sido cambiado, la posición de cambio y el nuevo valor). El uso del Redo Buffer es estrictamente secuencial, en tal sentido pueden entrelazarse cambios en los bloques de datos producidos por transacciones diferentes.

c) Area de SQL compartida (Share Pool)

Está compuesta, principalmente, de tres estructuras:

1. Dictionary Cache. Contiene la información del diccionario de datos usado recientemente. Cuando una información sobre un objeto (definición de columnas, usuarios, password, etc) se necesita, se leen las tablas del diccionario y su información se guarda en esta caché.
2. Library Cache.- Contiene la información de las sentencia SQL y PL/SQL ya ejecutada sobre la base de datos. Si un único usuario ejecuta una sentencia idéntica a otra ya ejecutada, se aprovecha el análisis de esa sentencia y se acelera su ejecución.
3. Result Cache.- Almacena el resultado de las consultas SQL o funciones PL/SQL (sólo está disponible en la versión Enterprice)

d) Cola de Petición y cola de Respuesta (Request Queue y Response Queues)

Cuando un proceso usuario desea establecer conexión con el manejador, la solicitud de conexión será encolada en la Request Queue. Los procesos servidores tomarán las solicitudes de conexión, efectuarán las acciones necesarias para que la base de datos complete la solicitud y colocarán la petición en la Response Queue asociada al proceso que atendió la solicitud.

e) Java Pool

Memoria usada por la máquina virtual Java Integrada

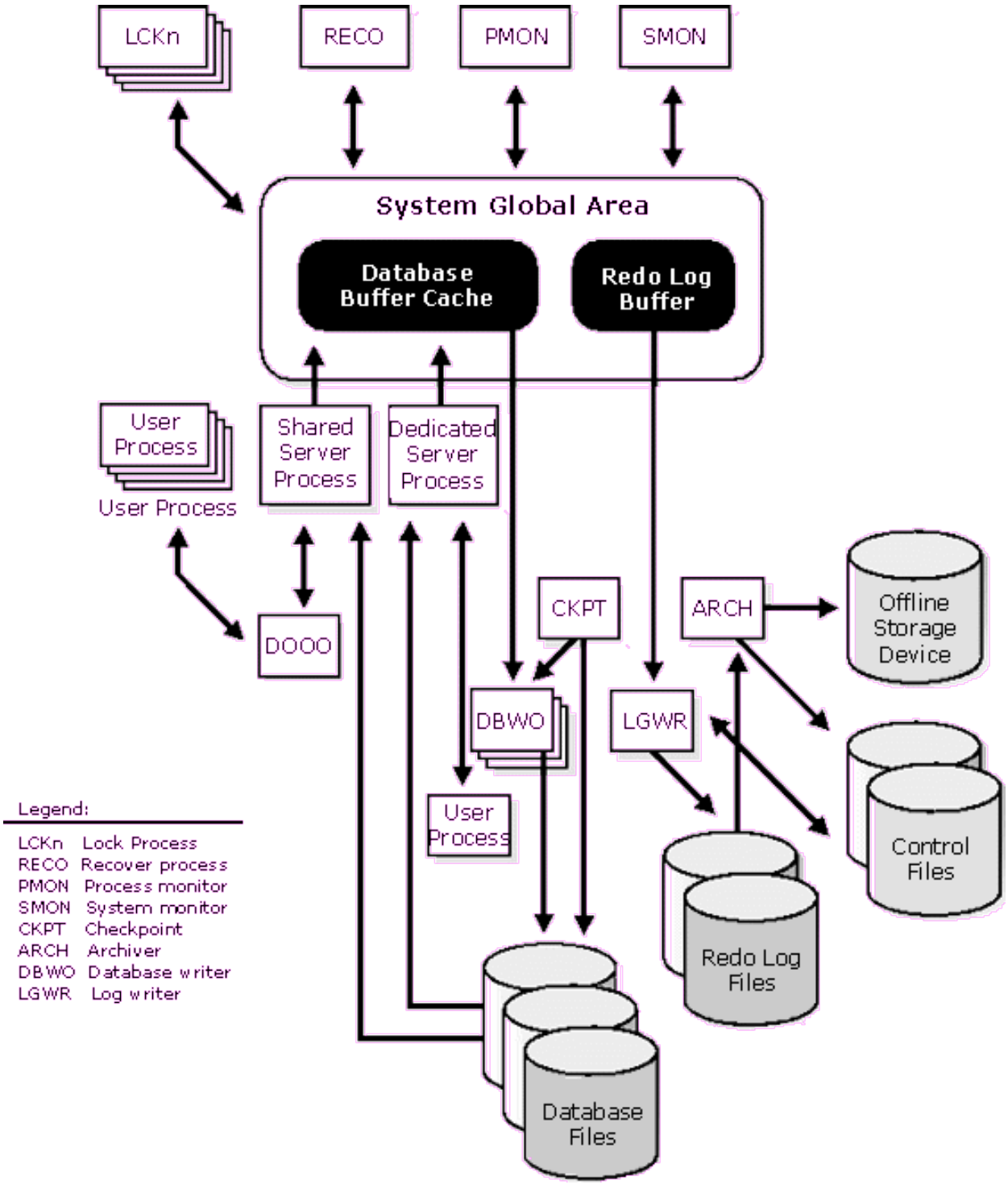
f) Large Pool

Es una zona de memoria opcional que es utilizada por diferentes procesos en algunas configuraciones

g) Stream Pool

Es una zonas de memoria que permite que la información circule entre procesos

5.- PROCESOS BACKGROUND DE LA BASE DE DATOS



Los procesos background más importantes son:

5.1.- Database Writer (DBWR):

Encargado de copiar los bloques de datos modificados desde el buffer cache hacia los archivos de la base de datos. Solamente trata los bloques que han cambiado (bloques sucios). El DBWR, escribe primero los bloques mas viejos (los usados menos recientemente). Estos bloques no se escriben necesariamente tras un commit de la transacción, lo único que sucede tras un commit de una transacción es un confirmación de que los cambios son registrados y escritos en los redolog on-line, Estos serán pasados a disco, cuando no hayan suficientes buffers libres en el SGA para un nuevo bloque. También se obliga la escritura tras un checkpoint.

5.2.- Log Writer (LGWR):

Encargado de escribir las entradas desde el Log Buffer a disco. Cuando se recibe un commit de una transacción, este proceso, debe escribir las entradas del buffer de redolog en los ficheros de redolog, antes de que el proceso reciba un mensaje indicando que el commit se ha realizado con éxito. Tras la confirmación, los datos ya están seguros en el disco, aunque los bloques se encuentren aún en el buffer cache, esperando a ser escritos por DBWR.

La escritura de bloques del Redo Log Buffer a disco ocurre secuencialmente.

5.3.-Checkpoint (CKPT)

Es el encargado de notificar al DBWR, para que se escriban en los archivos de datos todos los bloques contenidos en la lista de sucios. Este proceso es invocado en intervalos de tiempo determinados.

La secuencia de puntos de control se almacena en los ficheros de datos, redo logs y control para indicar el checkpoint mas reciente.

El CKPT es opcional. Si este proceso no está presente las funciones son realizadas por el LGWR.

5.4.- SMON (System Monitor)

Es el supervisor del sistema y se encarga de todas las recuperaciones que sean necesarias durante el arranque. Realiza la recuperación de la instancia a partir de los ficheros Redo Log, limpia los segmentos temporales no utilizados y compacta los huecos libres en los ficheros de datos. Se activa periódicamente para comprobar si debe intervenir.

5.5.- PMON (Process monitor)

Restaura las transacciones no validadas de los procesos de usuario que abortan, liberando los bloques y los recursos de la SGA de procesos de usuarios inactivos (muertos). Se activa de forma periódica para comprobar si debe intervenir.

5.6.- ARCH (Archiver)

Es opcional, sólo se utiliza cuando la base de datos se ejecuta en modo archive log. Es el encargado de realizar una copia a disco del archivo redo-log antes de reutilizarlo.

5.7.- RECO (Recover).

Está asociado a un servidor distribuido. Es opcional. Es el encargado de resolver transacciones distribuidas que se encuentran pendientes debido a la red o a fallos ocurridas en la base de datos distribuida.

5.8.- Dnnn (Dispatchers).

Es un proceso opcional, que está presente en el caso de configuraciones de servidor compartido. Al menos un proceso Distpachers es creado por cada uno delos protocolos de comunicación. Cada proceso Dnnn es responsable de enrutar la petición de la conexión de un proceso de usuario hacia el servidor compartido disponible, asegurando la respuesta hacia el proceso de cliente que lo solicitó

6.- MANEJANDO LOS TABLESPACES.

El manejo de las extensiones los tablespaces se puede llevar a cabo mediante dos formas:

- Manejado por el diccionario (Dictionary Manager). La información sobre su gestión se almacena en el diccionario de datos.
- Manejado localmente (Locally Manager). La información la gestión de su espacio se almacena en un bitmap, en el encabezado de cada fichero de datos del tablespace. De esta manera se reduce dramáticamente el acceso a las tablas del diccionario de datos. A partir de la versión 9i, esta es la que se utiliza por defecto (salvo el tablespace SYSTEM que es, por defecto, gestionado por el diccionario de datos).

Las Ventajas de la manejar los tablespace de forma local son :

- Reduce la contención sobre las tablas del catálogo (diccionario de datos)
- Rastrea automáticamente el espacio adyacente libre, eliminando la necesidad de unirlo.
- Evita operaciones recurrentes sobre las tablas del diccionario.
- El tamaño de los extensiones puede ser determinado automáticamente por system.
- Los cambios en los bitmaps no generan información de rollback, porque no actualizan las tablas en el diccionario. (excepto casos especiales como información de cuota del tablespace).
- Reduce la fragmentación

CREACIÓN DE TABLESPACE PERMANENTES

(manejado localmente).

CREATE [BIGFILE | SMALLFILE] TABLESPACE nombre_tablespace

DATAFILE 'nombre_archivo' SIZE entero[K | M | G | T] [REUSE] [cláusula_autoextend]

[, 'nombre_archivo' SIZE entero[K | M | G | T] [REUSE] [cláusula_autoextend]] ...

[EXTENT MANAGEMENT LOCAL {AUTOALLOCATE | UNIFORM SIZE entero [K|M|G|T]}]

[SEGMENT SPACE MANAGEMENT {MANUAL | AUTO}]

[ONLINE | OFFLINE]; // Tablespace disponible o no. Por defecto online

BIGFILE permite usar datafiles de tamaños muy grandes. Si no se indica esta opción cogerá por defecto el que esté definido en la base de datos

REUSE reutiliza el fichero si existe, en caso de que no exista lo crea.

Cláusula autoextend de los datafiles:

La cláusula AUTOEXTEND permite gestionar el crecimiento automático de los archivos de datos cuando estos se llenan. Su sintaxis es la siguiente:

AUTOEXTEND {OFF | ON [NEXT integer [K|M|G|T]] [MAXSIZE {UNLIMITED | integer [K|M|G|T]}]}

La opción por defecto es OFF

Next especifica el tamaño mínimo del siguiente bloque de espacio que Oracle reservará para el archivo de datos.

Maxsize indica el tamaño máximo que crecerá el archivo de datos

Manejo de extensiones en el tablespace:

EXTENT MANAGEMENT LOCAL {AUTOALLOCATE | UNIFORM SIZE entero [K|M|G|T]}

Indica que las extensiones se manejen localmente.

Oracle permite dos opciones para asignar el espacio de las extensiones:

- UNIFORM: asigna y desasigna el espacio en las extensiones de un tamaño uniforme igual a un valor definido en el momento de la creación de los tablespaces. Este es el valor por defecto en los tablespaces temporales y no se puede especificar en los tablespaces de undo.
- AUTOALLOCATE: El tamaño de las extensiones se determina automáticamente por Oracle. Su crecimiento se va realizando de forma proporcional. Es la que se usa por defecto

Manejo del espacio en los segmentos:

SEGMENT SPACE MANAGEMENT {MANUAL | AUTO}

Esta cláusula define el modo de gestión del espacio libre de los segmentos almacenado en los tablespace, si es automático usa un mapa de bits para gestionar el espacio libre, sin embargo si es manual usa “Free List”

Oracle recomienda encarecidamente que se use auto, que es la opción por defecto

MODIFICACIÓN TABLESPACES PERMANENTES

ALTER TABLESPACE nombre_tablespace

[RENAME nuevo_nombre_tablepace]

[ADD DATAFILE 'nombre_archivo' SIZE entero[K | M] [REUSE] [cláusula_autoextend][, 'nombre_archivo' ...]

[DROP DATAFILE 'nombre_archivo' | n_archivo]

[RENAME DATAFILE 'nombre_archivo' [, 'nombre_archivo2'] TO 'nnombre_archivo' [,'nnombre_archivo2']]

[ONLINE | OFFLINE];

El tablespace a renombrar y todos sus datafiles deben estar online.

No se pueden renombrar los tablespaces SYSTEM y SYSAUX.

CREACIÓN TABLESPACE TEMPORAL

CREATE [BIGFILE | SMALLFILE] TEMPORARY TABLESPACE nombre_tablespace

TEMPFILE ‘nombre_archivo’ SIZE entero[K | M | G | T] [REUSE]

[, ‘nombre_archivo’ SIZE entero[K | M | G | T] [REUSE], ...]

[cláusula_autoextend]

[EXTENT MANAGEMENT LOCAL [UNIFORM [SIZE entero[K | M | G | T]]]

Extent managent local uniform es la opcion que tiene por defecto, con un tamaño de 1M (es suficiente en la mayoria de los casos). Si se quiere otro valor hay que especificar esta claúsula especificando otro tamaño en size

Siempre es online

BORRADO TABLESPACES

DROP TABLESPACE tablespace [INCLUDING CONTENTS [{AND | KEEP} DATAFILES]

[CASCADE CONSTRAINTS]] ;

“including contents” se borra tablespace con datos

“keep datafiles” no borra los archivos de datos del S.O. Es la opción por defecto

“and datafiles” se borran los archivos de datos en el sistema de ficheros del S.O.

“cascade constraints” eliminar todas las restricciones de integridad referencial de tablas fuera del tablespace que hacen referencia a tablas dentro del tablespace a borrar.

Es recomendable que el tablespace este off-line.

MODIFICACIÓN DE DATAFILES

ALTER DATABASE DATAFILE ‘nombre_archivo’ [RESIZE tamaño [K|M]] [cláusula_autoextend]

[OFFLINE | ONLINE];

Algunas de las vistas del diccionario de datos para obtener información de los archivos de datos y tablespace son: dba_tablespaces y dba_data_files