

TEMA 6.

NET9.

TEMA 6. NET.

- ORACLE NET SERVICES.
- ORACLE NET.
- LISTENER.
- ORACLE CONNECTION MANAGER.
- OTROS COMPONENTES
- CONCEPTOS.
- CONFIGURACION.
- LISTENER.ORA
- LISTENER.ORA Y JSERVER.
- CONTROL DEL "LISTENER".
- CONFIGURACION BASICA NET.
- SQLNET.ORA
- DATABASE LINK.

ORACLE NET SERVICES.

- “Oracle Net Services” proporciona soluciones de conectividad para entornos distribuidos. Está compuesto por:
 - Oracle Net.
 - Listener.
 - Oracle Connection Manager.
 - Oracle Net Configuration Assistant.
 - Oracle Net Manager.

ORACLE NET.

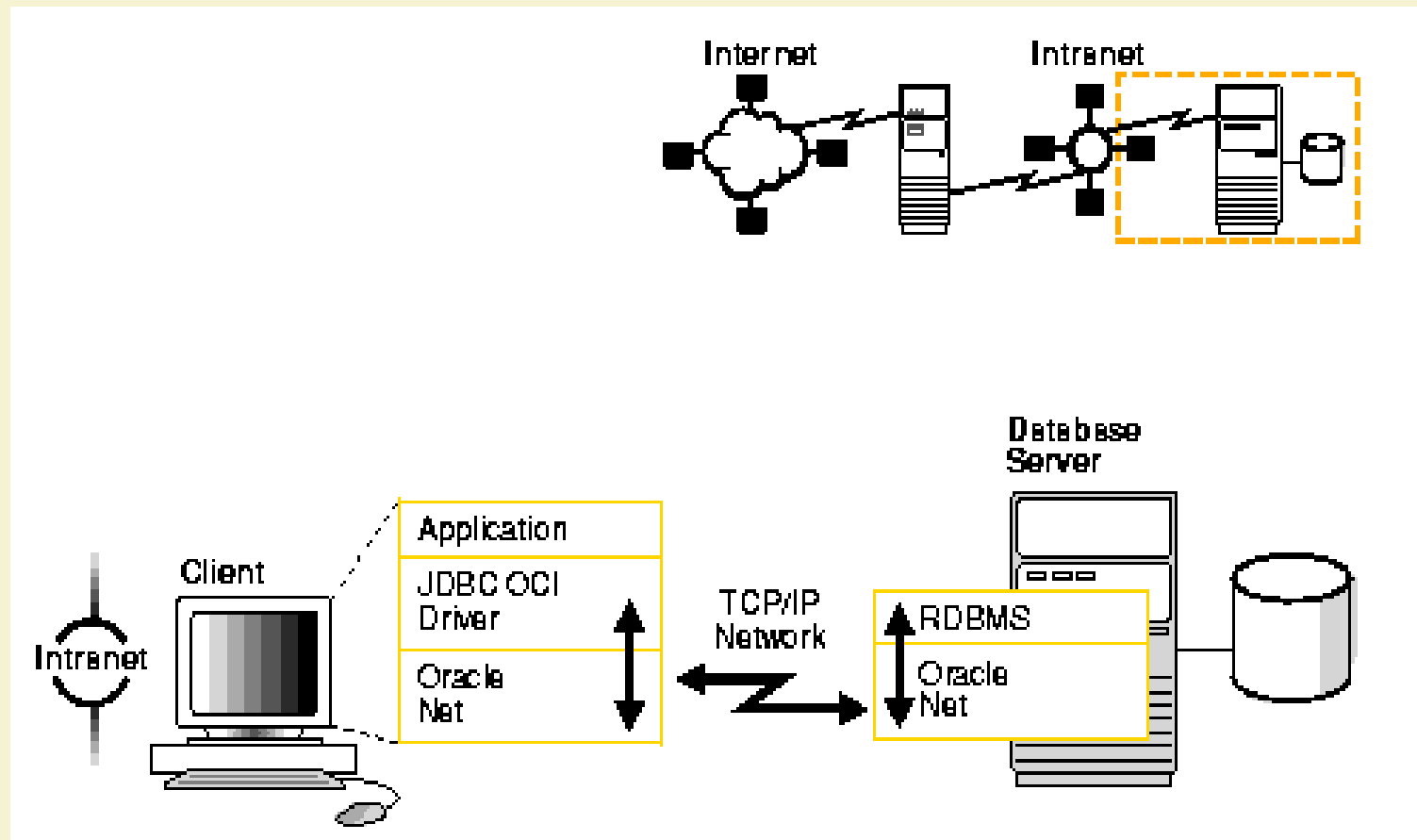
- Oracle Net es el sw que permite establecer y mantener una sesión de red desde una aplicación cliente a un servidor de bd; así como del intercambio de mensajes entre ambos.

Es capaz de hacerlo al estar localizado en cada ordenador de la red.

ORACLE NET.

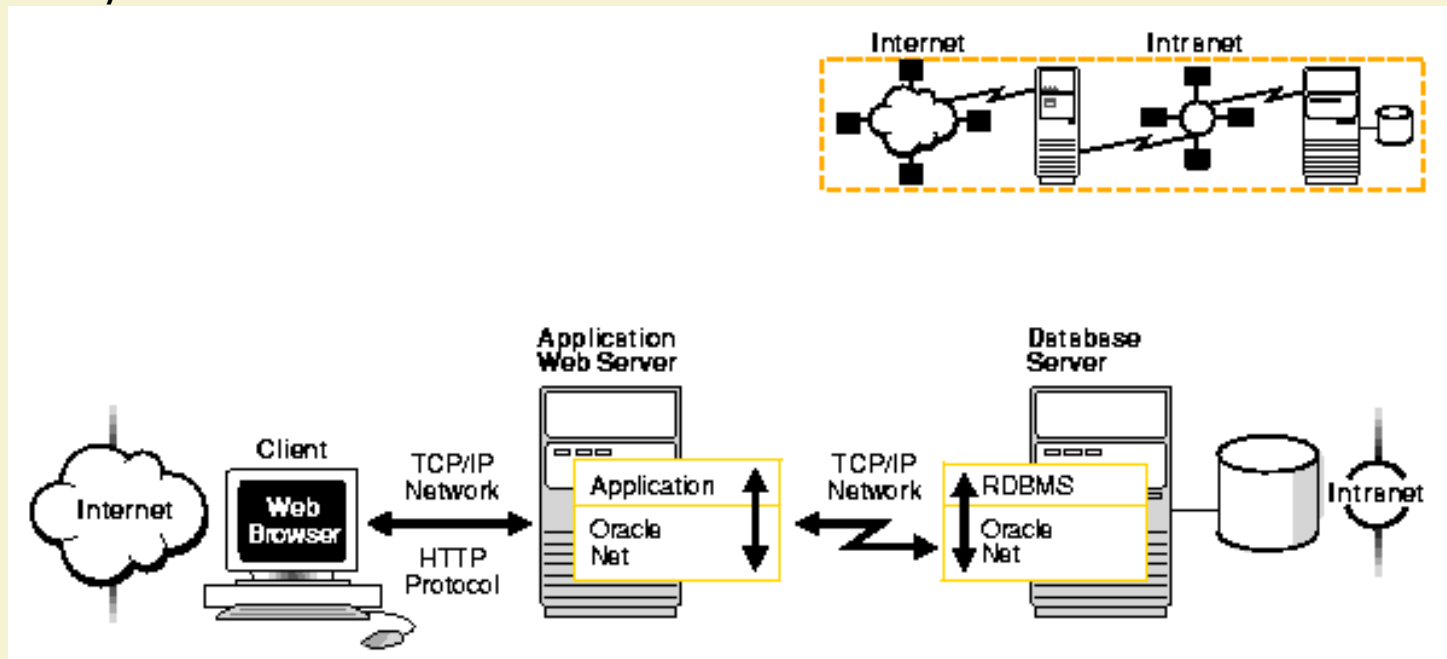
- Conexión de aplicaciones C/S.
 - Oracle Net reside en el cliente y en el servidor. Se sitúa en la capa más alta de protocolos de red, que indica como las aplicaciones acceden a la red y como los datos se dividen en paquetes para su transmisión.
 - Oracle Net se compone a su vez de "Oracle Net foundation layer" (establece y mantiene la conexión) y " Oracle protocol support" (mapea a protocolos estándar desde la tecnología Oracle Net).
 - Las aplicaciones cliente Java acceden a la bd Oracle a través de JDBC (Java Database Connectivity Driver). Usan Oracle Net para comunicar con la bd.

ORACLE NET.



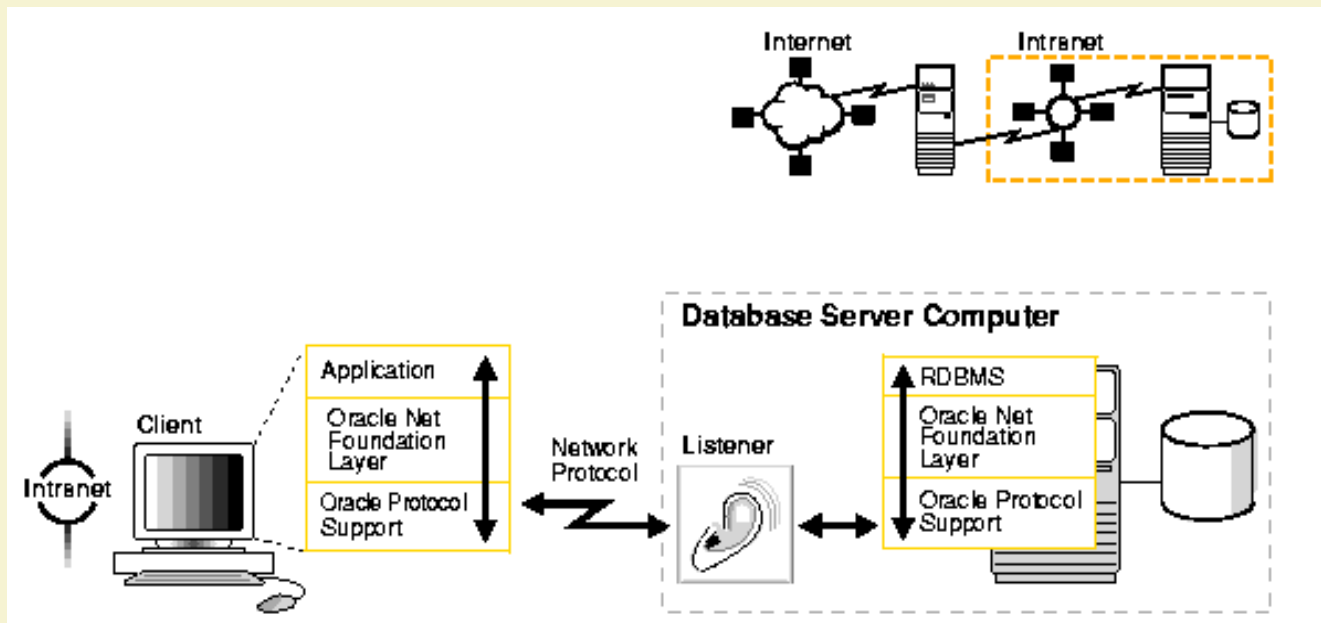
ORACLE NET.

- Conexión de aplicaciones web cliente (Con o sin Web Application Server).
 - La conexión desde navegador a una bd es similar al caso C/S.



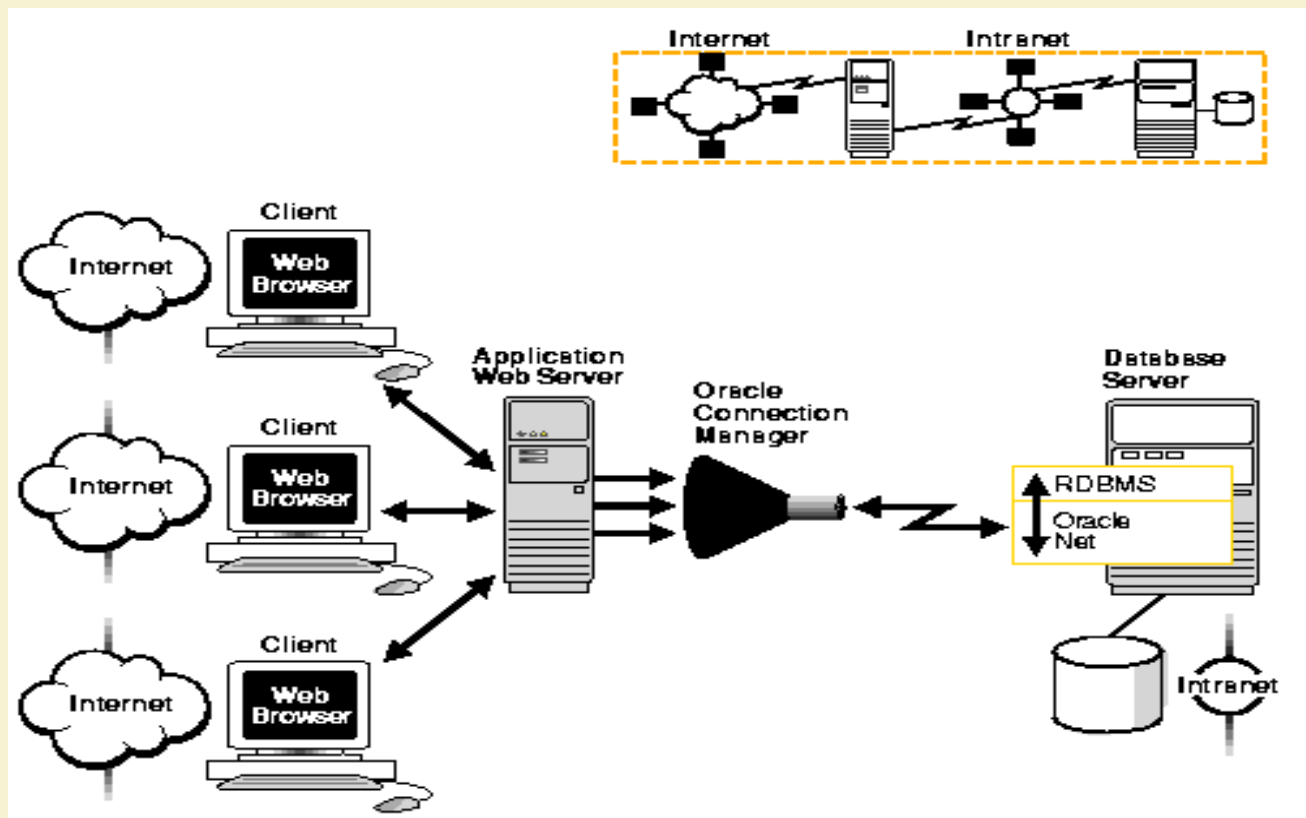
LISTENER.

- En la bd recibe la conexión inicial. Está configurado con una dirección de protocolo, aquellos clientes configurados con esta dirección pueden enviar peticiones al "listener". Una vez establecida la conexión, el cliente y la bd comunican directamente entre ellos.



ORACLE CONNECTION MANAGER.

- Permite que múltiples sesiones cliente sean multiplexadas a través de una sola conexión a una bd.



OTROS COMPONENTES

- Oracle Net Configuration Assistant.
 - Herramienta que permite configurar los componentes básicos de red después de realizar la instalación de bd.
 - Oracle Universal Installer lanza esta herramienta durante la instalación del sistema gestor de bd.
 - En UNIX debe ejecutarse `$ORACLE_HOME/bin/netca`

- Oracle Net Manager.
 - Proporciona un entorno integrado de configuración y gestión de Oracle Net Services (En UNIX debe ejecutarse `$ORACLE_HOME/bin/netmgr`).

CONCEPTOS.

- SERVICIO.
 - Una bd es un "servicio" desde el punto de vista de los clientes.
 - El nombre de servicio viene determinado por el parámetro de inicialización `SERVICE_NAMES`. Por defecto es el nombre "global database", compuesto por el nombre de bd (parámetro `DB_NAME`) y el nombre de dominio (`DB_DOMAIN`).
 - Este parámetro también puede ser modificado dinámicamente mediante `ALTER SYSTEM`.
 - La bd puede tener uno o más servicios asociados; así puede ser identificada de distintas formas por clientes diferentes, y un administrador puede limitar recursos del sistema.

CONCEPTOS.

- INSTANCIA.
 - El nombre de la instancia viene indicado por el parámetro de inicialización `INSTANCE_NAME`. Por defecto se refiere al Oracle System Identifier (SID) de la instancia.

CONCEPTOS.

- CONECTOR.

- Un cliente usa un descriptor de conexión que indica la localización de la bd y el nombre del servicio de bd.

```
(DESCRIPTION=  
  (ADDRESS=(PROTOCOL=tcp)  
    (HOST=<nombre/direccionIP>)(PORT=1521))  
  (CONNECT_DATA=  
    (SERVICE_NAME=<nombre_de_servicio>)))
```

También puede indicarse la conexión a una instancia específica (Real Application Clusters):

```
(INSTANCE_NAME=<nombre_instancia>)))
```

CONFIGURACION.

- Se considera un modelo de configuración **no centralizado**, donde la información sobre direcciones de red se almacena en el fichero *tnsnames.ora* de cada sistema.
- Existen los siguientes ficheros de configuración:
 - *listener.ora* ... En el servidor de bd. Con información sobre protocolo , dirección de escucha, servicios para los que se escucha y parámetros de control.
 - *sqlnet.ora* ... En el cliente y en el servidor bd. Puede contener ruta de conexiones, parámetros de control de acceso a bd, parámetros de Oracle Advanced Security, ...
 - *tnsnames.ora* ... En los clientes. Contiene "net service names" -nombre de servicio asociado a un descriptor de conexión-.

CONFIGURACION.

- Los ficheros de configuración residen generalmente en *\$ORACLE_HOME/network/admin* . El orden de búsqueda para *sqlnet.ora* es:
 - El directorio indicado por la variable *TNS_ADMIN*
 - El directorio *\$ORACLE_HOME/network/admin*
- El orden de búsqueda para *listener.ora* y *tnsnames.ora* es:
 - El directorio indicado por la variable *TNS_ADMIN*
 - En UNIX, el directorio global de configuración ("global configuration directory").
 - El directorio *\$ORACLE_HOME/network/admin*

LISTENER.ORA

- El "listener" es un proceso que corre en la bd. Recibe peticiones de conexión de los clientes y gestiona el tráfico de dichas peticiones a la bd.
- Su configuración se almacena en un fichero llamado listener.ora. Dado que todos los parámetros de configuración tienen valores por defecto, es posible arrancar y usar el "listener" sin configurar. El "listener" se llama por defecto LISTENER y escucha atendiendo a los siguientes valores:

(ADDRESS=(PROTOCOL=tcp)(HOST=host_name)(PORT=1521))

LISTENER.ORA

- Durante la instalación de la bd, Oracle Universal Installer ejecuta el Oracle Net Configuration Assistant; este permite configurar el "listener" con un nombre y una dirección concreta.
- También se crea una entrada para escucha del protocolo IPC que se configura de forma automática y sirve a procedimientos externos (escritos en un 3GL que pueden ser llamados desde el código PL/SQL -sólo C-).

LISTENER.ORA

- En la configuración del "listener" se indica:
 - Nombre.
 - Dirección que acepta peticiones.
 - Servicios para los que escucha (la configuración estática es necesaria si se usa Oracle Enterprise Manager).
 - Parámetros de control.
- Pueden configurarse múltiples "listener", con distintos nombres, en un fichero listener.ora. Sin embargo, Oracle recomienda ejecutar sólo uno por nodo en la mayoría de entornos.

LISTENER.ORA

- Sección Protocolo.
- Define las direcciones de la que el "listener" acepta peticiones.

```
LISTENER=  
(DESCRIPTION=  
  (ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=tcp)(HOST=sale-server)  
      (PORT=1521))  
    (ADDRESS=(PROTOCOL=ipc)(KEY=extproc))))
```

- El puerto por defecto para conexiones es el 1521 -en un futuro puede cambiar a 2483, TCP/IP, y 2484, TCP/IP con SSL-. Puertos inferiores a 1024 están reservados para su uso por procesos privilegiados.

LISTENER.ORA

- El parámetro *QUEUESIZE* permite indicar el número de peticiones concurrentes que puede aceptar el “listener” via TCP/IP. Su valor por defecto depende del s.o..

```
listener_name=  
(DESCRIPTION=  
(ADDRESS=(PROTOCOL=tcp)(HOST=hr-server)  
(PORT=1521)(QUEUESIZE=20)))
```

LISTENER.ORA

- Sección Registro Servicio (SID_LIST).
- Requerida por compatibilidad con bdd Oracle8, procedimientos externos y ciertas herramientas de gestión (Oracle Enterprise Manager). En Oracle9i u 8i está información se registra de forma dinámica al arrancar la instancia.

```
SID_LIST_<nombre_listener> =  
(SID_LIST=  
  (SID_DESC=  
    (GLOBAL_DBNAME=sales.us.acme.com)  
    (ORACLE_HOME=/oracle9i)  
    (SID_NAME=sales))  
  (SID_DESC=  
    (SID_NAME=plsextproc)  
    (ORACLE_HOME=/oracle9i)  
    (PROGRAM=extproc)))
```

LISTENER.ORA

- Donde:
 - *SID_LIST ... lista de descripciones*
 - *SID_DESC ... información de servicio*
 - *GLOBAL_DBNAME ... "Global database name" de la bd (parámetro SERVICE_NAMES).*
 - *ORACLE_HOME ... localización del servicio (opcional en UNIX).*
 - *PROGRAM ... Nombre del programa de servicio ejecutable*
 - *SID_NAME ... SID de la instancia (parámetro INSTANCE_NAME)*

LISTENER.ORA

- Parámetros de control.
 - *LOG_DIRECTORY_ <nombre_listener>* indica el directorio de destino del fichero de log del "listener". Por defecto el directorio *\$ORACLE_HOME/network/log*
 - *LOG_FILE_ <nombre_listener>* indica el nombre del fichero de log para el "listener". Por defecto *listener.log*
 - *TRACE_DIRECTORY_ <nombre_listener>* indica el directorio de destino de los ficheros de traza del "listener". Por defecto *\$ORACLE_HOME/network/trace*
 - *TRACE_FILE_ <nombre_listener>* indica el nombre del fichero de traza del "listener". Por defecto *listener.trc*

LISTENER.ORA Y JSERVER.

- Clientes acceden a aplicaciones Enterprise JavaBeans (EJBs) y Common Object Request Broker Architecture (CORBA), opción Oracle Jserver, en una bd Oracle8i sobre una conexión Inter-Orb Protocol (IIOP). IIOP es una implementación de General Inter-Orb Protocol (GIOP) sobre TCP/IP. Debe configurarse el puerto 2481 para TCP/IP o 2482 para TCP/IP con SSL.

```
listener=  
  (DESCRIPTION_LIST=  
    (DESCRIPTION=  
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-server)  
(PORT=2481))  
      (PROTOCOL_STACK=  
        (PRESENTATION=giop)  
        (SESSION=raw))))
```


CONTROL DEL "LISTENER".

- La utilidad para control del "listener" (Listener Control Utility) permite administrar este componente. La sintaxis es:

```
S.O.> LSNRCTL <sentencia> [nombre_listener]
```

- Si no se indica nombre, hace referencia al "listener" de nombre "listener". Para **arrancar** el "listener" lsnr:

```
LSNRCTL START lsnr
```

- También pueden ejecutarse sentencias en el "prompt" LSNRCTL (se obtiene tecleando lsnrctl sin argumentos).

```
S.O.> lsnrctl
```

```
LSNRCTL> START lsnr_HOME/network/admin
```

CONTROL DEL "LISTENER".

- Pueden combinarse distintas órdenes en un fichero de texto y ejecutarlas de la forma:

lsnrctl @file_name

- Para **parar** el "listener" lsnr:

LSNRCTL STOP lsnr

- Para **obtener información** del "listener"

LSNRCTL STATUS lsnr

CONTROL DEL "LISTENER".

- Mediante la orden SERVICES se obtiene información sobre los servicios e instancias registradas, y los manejadores asociados a cada instancia.

LSNRCTL SERVICES Isnr

CONFIGURACION BASICA NET.

- Se muestra la forma de intercomunicar una aplicación cliente con una bd a través de una red TCP/IP.
- Se asume que:
 - BD y Cliente en la misma red.
 - BD: Instalado TCP/IP y configurado un “listener”.
 - Cliente: Instalado Oracle Client e instalado TCP/IP.

CONFIGURACION BASICA NET.

1.- Confirmar conexión de red.

- Realizar un "loopback test" para comprobar que la bd puede comunicar con si misma

ping <nombre_maquina> o ping <direccion_IP>

- Comprobar que existe comunicación entre el cliente y la bd (por ejemplo, con "ping").

CONFIGURACION BASICA NET.

2.- Arrancar el "listener" y la bd.

- Desde s.o.: *lsnrctl start <nombre_listener>*

Nombre_listener debe estar definido en el fichero listener.ora

- Arrancar la bd.

sqlplus /nolog

SQL> CONNECT username/password as sysdba

SQL> STARTUP database_name pfile=file

CONFIGURACION BASICA NET.

- Comprobar que el "service registration" de la bd con el "listener" se ha llevado a cabo (utilidad por la que el proceso PMON automáticamente proporciona información al "listener" tal como el nombre de servicio para cada instancia de la bd, nombres de instancias de la bd, y manejadores -"dispatchers" o servidores dedicados- disponibles para la instancia, ...; que permite, entre otras cosas, redirigir una petición cliente adecuadamente).

```
LSNRCTL> SERVICES [nombre_listener]
```

CONFIGURACION BASICA NET.

3.- Configurar el cliente para usar un nombre de servicio de red ("Net Service Name").

Se trata de un nombre para el servicio asociado a un descriptor de conexión. El cliente, para conectarse, indicará un nombre de usuario, una contraseña y una cadena de conexión o nombre de servicio (reside en el fichero *tnsnames.ora* -modificable manualmente, con NetManager, o con NetCAssistant-). Por ejemplo:

```
sales=  
(DESCRIPTION=  
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)  
    (PORT=1521))  
  (CONNECT_DATA= (SERVICE_NAME=sales.us.acme.com)))
```


CONFIGURACION BASICA NET.

- El descriptor de conexión contiene:
 - La ruta de red hacia la bd, incluyendo la localización del "listener" a través de un protocolo -sección ADDRESS-.
 - Nombre de servicio ("service name") de una bd versión 8i o 9i, o el SID de la bd en caso de versiones 8.0 -sección CONNECT_DATA-.

Puede indicarse una instancia con *INSTANCE_NAME*.

```
sales=  
(DESCRIPTION=  
(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521))  
(CONNECT_DATA= (SERVICE_NAME=sales.us.acme.com)  
(INSTANCE_NAME=sales)))
```

CONFIGURACION BASICA NET.

- Al crear un descriptor para una bd versión 8.0, el servicio se identifica con el parámetro SID.

```
sales=  
(DESCRIPTION=  
(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)  
(PORT=1521))  
(CONNECT_DATA= (SID=sales)))
```

CONFIGURACION BASICA NET.

- Para configurar un modelo local o **no centralizado**, el parámetro *NAMES.DIRECTORY_PATH* del fichero *sqlnet.ora* -indica el orden de los métodos de configuración que Oracle Net emplea para resolver los identificadores- debe contener en primer lugar el valor *tnsnames*.

NAMES.DIRECTORY_PATH=(tnsnames)

CONFIGURACION BASICA NET.

4.- Conectar a bd.

- Desde el cliente:
 - Arrancar SQL*Plus: *sqlplus*
 - Conectar a bd:

CONNECT username/password@nombre_servicio

SQLNET.ORA

- Los parámetros de configuración para cliente y servidor, perfil, que permiten especificar preferencias de Oracle Net se almacenan en el fichero sqlnet.ora
- Permite realizar acciones como:
 - Priorizar modelos de localización (*NAMES.DIRECTORY_PATH*).
 - Permitir o no el acceso a ciertos clientes.
 - Acceso denegado.
TCP.EXCLUDED_NODES=(<dir_IP> | <servidor>)
 - Acceso permitido:
TCP.INVITED_NODES=(<dir_IP> | <servidor>)
 - Comprobar valores anteriores .
TCP.VALIDNODE_CHECKING = yes/no

SQLNET.ORA

- Indicar el intervalo, en segundos, que media para el envío de una confirmación para verificar que la conexión está activa.
 - *SQLNET.EXPIRE_TIME*. Un valor mayor que 0 asegura que no permanecen sesiones abiertas indefinidamente. Si la confirmación encuentra una sesión que ha terminado o que no está en uso, devuelve un error y fuerza al proceso servidor a terminar.

DATABASE LINK.

- Es una conexión entre dos bdd que permite acceder a ellas como una única bd; así como acceder de forma remota. Define una vía de comunicación unidireccional, en un solo sentido, cuya definición se almacena en diccionario de datos.
- Pueden ser públicos o privados.
- Generalmente tienen el mismo nombre que el "global database name" de la bd remota que referencia.

TEMA 7.

ARQUITECTURA "SHARED SERVER".

TEMA 7. SHARED SERVER (MTS).

- SERVIDOR DEDICADO Y SERVIDOR COMPARTIDO.
- VENTAJAS "SHARED SERVER".
- "DISPATCHER".
- NUMERO DE "DISPATCHER".
- PROCESOS SERVIDOR COMPARTIDOS.
- OTROS PARAMETROS.
- MODIFICAR NUMERO "DISPATCHERS".
- VERIFICAR CONFIGURACION.
- PARAR "DISPATCHERS".
- CAMBIAR NUMERO "SHARED SERVER".
- FORZAR CONFIGURACION COMPARTIDA.
- FORZAR CONFIGURACION DEDICADA.
- VISTAS.

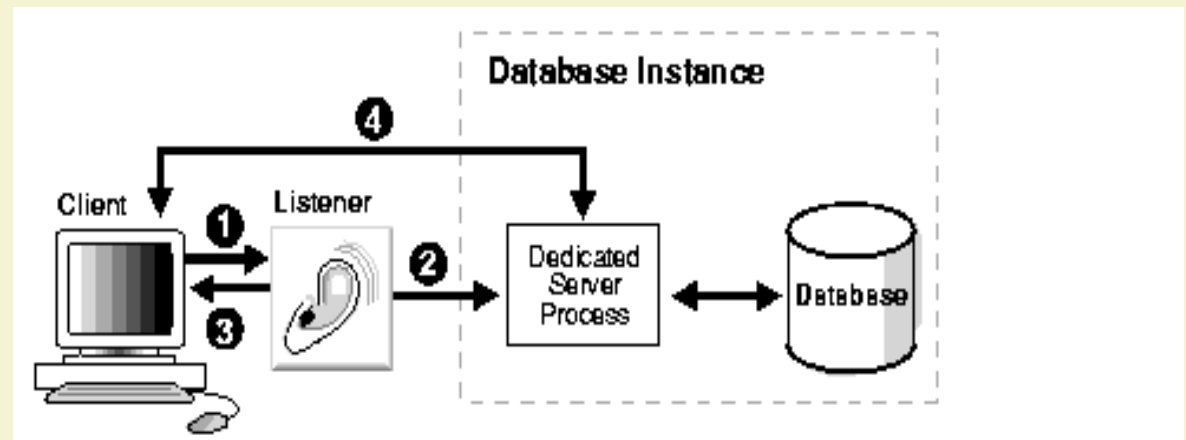
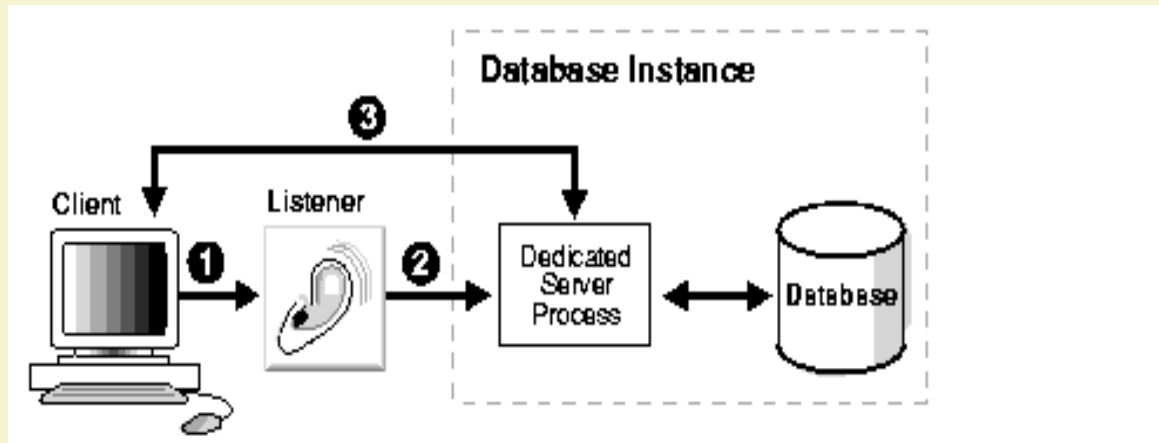
MANEJADORES DE SERVICIO.

- Para cada instancia existen unos “manejadores de servicio” o puntos de conexión a la bd.
- Estos pueden ser:
 - Un servidor dedicado, o
 - Un “dispatcher”, caso de servidor compartido.

SERVIDOR DEDICADO.

- El "listener" arranca un proceso dedicado exclusivo para cada petición cliente, que permanece hasta la finalización de la misma. Son necesario así más recursos que en el caso de un servidor compartido.
- Pueden darse dos casos:
 - El servidor dedicado hereda la petición de conexión del "listener".
 - El servidor dedicado informa al "listener" sobre su dirección de escucha, esta se pasa al cliente. Se cierra la conexión con el "listener" y el cliente se conecta al servidor dedicado directamente.

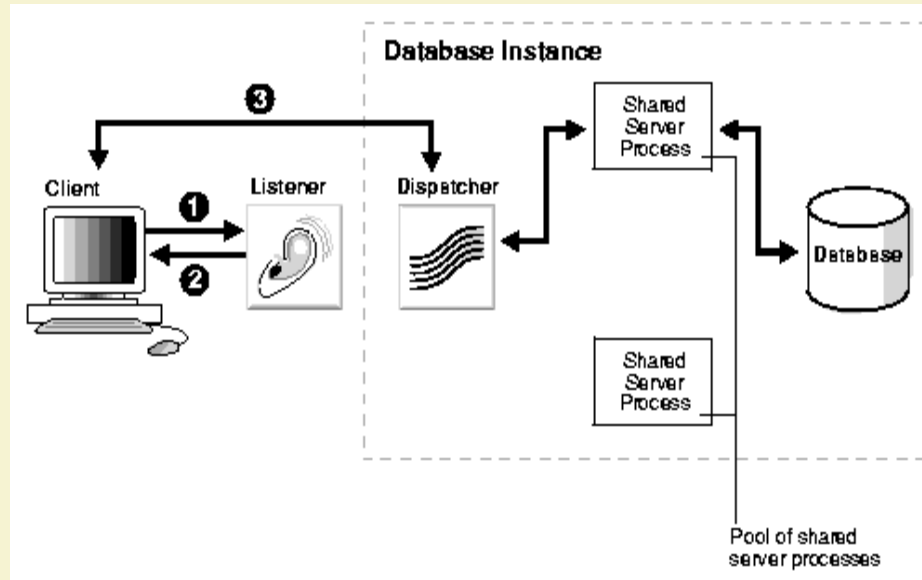
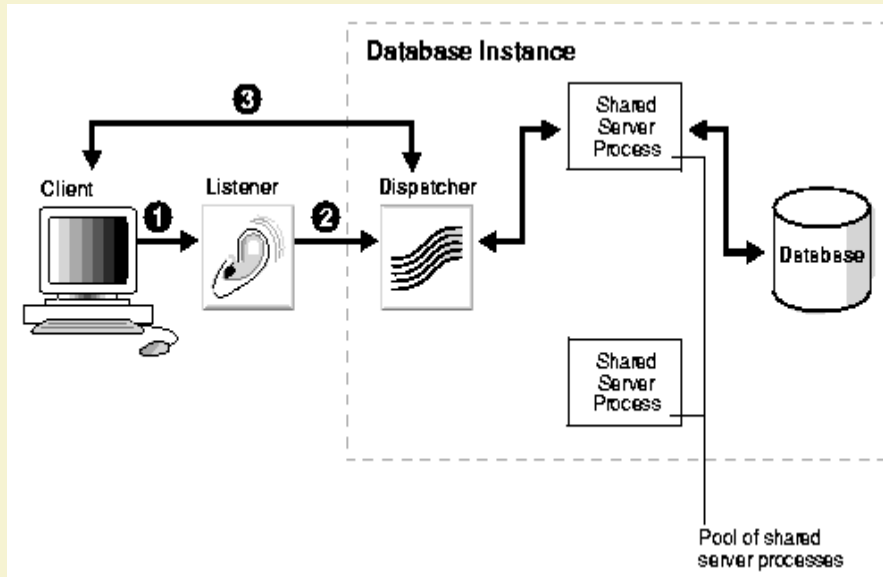
SERVIDOR DEDICADO.



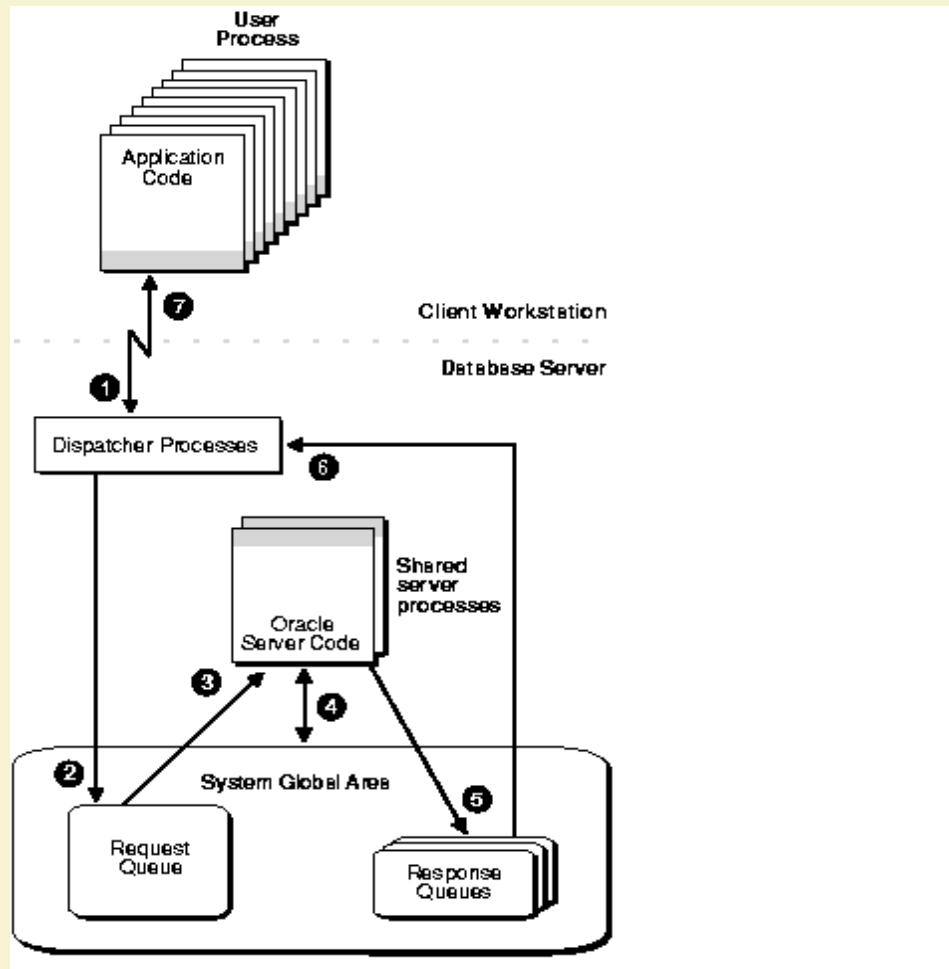
SERVIDOR COMPARTIDO: "DISPATCHER".

- La arquitectura de servidor compartido usa un proceso "dispatcher" para redirigir las conexiones cliente a una cola de peticiones común. Un proceso servidor desocupado extrae la petición de esta cola.
Permite que un pequeño número de procesos servidor atienda un gran número de clientes, reduciendo así los recursos de sistema necesarios.
- El proceso "listener" redirige las peticiones al "dispatcher". Al llegar una petición, el "listener" la conduce al "dispatcher", o indica al cliente la dirección de protocolo del "dispatcher" -el cliente termina la sesión con el "listener" y establece una con el "dispatcher"-.

SERVIDOR COMPARTIDO: "DISPATCHER".



SERVIDOR COMPARTIDO: "DISPATCHER".



VENTAJAS "SHARED SERVER".

- Reduce el número de procesos que corren sobre la instancia.
- Incrementa el número de posibles usuarios.
- Disminuye el número de procesos servidor ociosos.
- Reduce el uso de memoria y la sobrecarga del sistema.

“DISPATCHER”.

- El número de procesos “dispatcher” arrancados con la instancia se controla con el parámetro de inicialización *DISPATCHERS*. Este número depende de la bd, del límite impuesto por s.o., del número de conexiones para cada proceso, y del número de conexiones requeridas para cada protocolo.
- La instancia debe proporcionar tantas conexiones como usuarios concurrentes en la bd, teniendo en cuenta que tras arrancar la instancia pueden arrancarse más procesos si son necesarios. Una proporción apropiada es la de un “dispatcher” por cada 1000 conexiones, redondeando al entero superior.

“DISPATCHER”.

- Cada “dispatcher” puede aceptar hasta 1024 conexiones; este valor máximo es también el que se emplea por defecto.

Puede ajustarse a un número máximo de conexiones usando el parámetro CONN al configurar el parámetro DISPATCHERS en el fichero de parámetros de inicialización, o al emplear ALTER SYSTEM SET.

- Configurar demasiados “dispatcher” puede afectar a la productividad de la bd.

NUMERO DE "DISPATCHER".

- Configuración Típica.

```
DISPATCHERS="(PROTOCOL=TCP) (DISPATCHERS=2)"
```

```
DISPATCHERS="(PROTOCOL=IPC) (DISPATCHERS=1)"
```

- Forzar los puertos usados.

```
DISPATCHERS="(ADDRESS=(PROTOCOL=TCP)(PORT=5000))"
```

```
DISPATCHERS="(ADDRESS=(PROTOCOL=TCP)(PORT=5001))"
```

PROCESOS SERVIDOR COMPARTIDOS.

- El número de procesos servidor creados en el arranque se indica mediante el parámetro -dinámico- de inicialización *SHARED_SERVERS*.

SHARED_SERVERS = 5

- Si la carga del sistema decrece, se mantiene este mínimo por lo que no debe asignarse un valor demasiado alto. Por defecto, si se usa la arquitectura "shared server" el valor es 1; en caso contrario 0.
- Tanto el número de *DISPATCHERS* como de *SHARED_SERVERS* pueden modificarse dinámicamente. Es necesario el privilegio *ALTER SYSTEM*.

OTROS PARAMETROS.

- *CIRCUITS*. Parámetro estático que indica número total de circuitos virtuales disponibles para sesiones de red.
Un circuito virtual es una porción de memoria compartida usada por el "dispatcher" para las conexiones cliente (peticiones/respuestas). El "dispatcher" coloca un circuito virtual en una cola común cuando llega una petición; un servidor compartido ocioso coge dicho circuito, sirve la petición, y enlaza el circuito antes de intentar recuperar otro.
- *MAX_DISPATCHERS*. Parámetro estático que indica el máximo número de procesos "dispatcher" permitidos a la vez. Su valor por defecto es de 5.
- *MAX_SHARED_SERVERS*. Indica el número máximo de procesos de servidor compartido ("shared server") simultáneos. Si ocurren con demasiada frecuencia "deadlocks", debe incrementarse.

OTROS PARAMETROS.

- *SHARED_SERVER_SESSIONS*. Parámetro estático que indica el número total de sesiones concurrentes de usuario permitidas en arquitectura "shared server". Su valor oscila entre cero y *SESSIONS*-5; siendo su valor por defecto el menor de *CIRCUITS* y *SESSIONS* - 5.
- Parámetros que también pueden requerir ajuste son:
 - *LARGE_POOL_SIZE*. Tamaño en bytes de la pila "large pool", mínimo de 300k. Si no se indica un valor se emplea la "shared pool" para las sesiones de usuario -; el emplear la "large pool" hace decrecer la fragmentación de la "shared pool".
 - *SESSIONS*. Número máximo de sesiones a crear en el sistema-.

MODIFICAR NUMERO "DISPATCHERS".

- Si consultando las vistas *V\$QUEUE*, *V\$DISPATCHER* y *V\$DISPATCHER_RATE* se observa que la carga de los procesos "dispatcher" es alta, puede mejorar la productividad el arrancar más procesos. Si la carga es baja, debe reducirse el número de procesos.
- Para modificar el número de procesos se usa la sentencia *ALTER SYSTEM*. El número de procesos "dispatcher" puede elevarse hasta alcanzar *MAX_DISPATCHERS*. Cuando se reduce, puede hacerse hasta el número indicado en *DISPATCHERS*.

```
ALTER SYSTEM  
SET DISPATCHERS =  
'(PROTOCOL=TCP)(DISPATCHERS=5)',  
'(PROTOCOL=TCPS)(DISPATCHERS=2)';
```

CLAUSULA INDEX.

- INDEX. Este parámetro se usa en ALTER SYSTEM SET DISPATCHERS para indicar que "dispatcher" quiere modificarse (se ignora si se indica en el fichero de parámetros). Señala el orden en el que se inicializaron, su rango oscila entre 0 (el primer proceso "dispatcher") a uno menos al número total de "dispatchers" definidos.

Por ejemplo, si se indican 3 "dispatchers" en el fichero de parametros, para modificar el tercero debe indicarse INDEX=2 en la sentencia ALTER SYSTEM.

- Si en el fichero de inicialización se indica:

```
dispatchers = "(ADDRESS=(PROTOCOL=TCP) (host=indl040ad)
(port=5100))"
```

```
dispatchers = "(ADDRESS=(PROTOCOL=TCP) (host=indl040ad)
(port=2480))"
```


CLAUSULA INDEX.

- Ejecutando "lsnrctl services" (lsnrctl services listener_v9205):

Service "v9205" has 2 instance(s).

Instance "v9205", status UNKNOWN, has 1 handler(s) for this service...

Handler(s):

"DEDICATED" established:0 refused:0

LOCAL SERVER

Instance "v9205", status READY, has 3 handler(s) for this service...

Handler(s):

"DEDICATED" established:0 refused:0 state:ready

LOCAL SERVER

"D001" established:0 refused:0 current:0 max:1002 state:ready

DISPATCHER <machine: INDL040AD, pid: 840>

(ADDRESS=(PROTOCOL=tcp)(HOST=indl040ad.idc.oracle.com)

(PORT=2480))

"D000" established:0 refused:0 current:0 max:1002 state:ready

DISPATCHER <machine: INDL040AD, pid: 2280>

(ADDRESS=(PROTOCOL=tcp)(HOST=indl040ad.idc.oracle.com)

(PORT=5100))

The command completed successfully

CLAUSULA INDEX.

- Asignemos ahora el puerto 3500 al "dispatcher" D000's:
- SQL> alter system shutdown immediate 'D000';
System altered.

```
SQL> alter system set DISPATCHERS='(ADDRESS= PROTOCOL=TCP)
(HOST=indl040ad)(PORT= 3500))(INDEX = 0)';
System altered.
```

```
SQL> alter system register;
System altered.
```

•*Nota: Indicando REGISTER, el PMON registra la instancia con los "listeners" inmediatamente; en caso contrario los clientes pueden no acceder a los servicios temporalmente.*

•*Nota: También podría añadirse un "dispatcher" con ALTER SYSTEM indicando INDEX=3.*

CLAUSULA INDEX.

- Ejecutando "lsnrctl services" de nuevo:

Service "v9205" has 2 instance(s).

Instance "v9205", status UNKNOWN, has 1 handler(s) for this service...

Handler(s):

"DEDICATED" established:0 refused:0

LOCAL SERVER

Instance "v9205", status READY, has 3 handler(s) for this service...

Handler(s):

"DEDICATED" established:0 refused:0 state:ready

LOCAL SERVER

"D001" established:0 refused:0 current:0 max:1002 state:ready

DISPATCHER <machine: INDL040AD, pid: 840>

(ADDRESS=(PROTOCOL=tcp)(HOST=indl040ad.idc.oracle.com)

(PORT=2480))

"D000" established:0 refused:0 current:0 max:1002 state:ready

DISPATCHER <machine: INDL040AD, pid: 2280>

(ADDRESS=(PROTOCOL=tcp)(HOST=indl040ad.idc.oracle.com)

(PORT=3500))

The command completed successfully

VERIFICAR CONFIGURACION.

- Al usar "shared server", debe arrancarse en primer lugar el "listener" y después la bd. Puede verificarse el estado del "dispatcher" ejecutando:

lsnrctl services

- Para verificar que las conexiones realizadas están usando servidores compartidos debe consultarse la vista V\$CIRCUIT (habrá una entrada por cada conexión de servidor compartido).

Select dispatcher, circuit, server, status from v\$circuit;

PARAR "DISPATCHERS".

- Es posible parar procesos específicos al estar identificados por un nombre de la forma *Dnnn* -los procesos "background" correspondientes se identifican de la forma *Dnnn-*.

```
SELECT NAME, NETWORK FROM V$DISPATCHER;
```

```
NAME NETWORK
```

```
-----  
D000 (ADDRESS=(PROTOCOL=tcp)(HOST=rbaylis-hpc.us.oracle.com)  
      (PORT=3499))  
D001 (ADDRESS=(PROTOCOL=tcp)(HOST=rbaylis-hpc.us.oracle.com)  
      (PORT=3531))  
D002 (ADDRESS=(PROTOCOL=tcp)(HOST=rbaylis-hpc.us.oracle.com)  
      (PORT=3532))
```

- Para parar el "dispatcher" D002:

```
ALTER SYSTEM SHUTDOWN IMMEDIATE 'D002';
```

PARAR "DISPATCHERS".

- La cláusula *IMMEDIATE* permite parar el "dispatcher", no aceptando nuevas conexiones y terminando inmediatamente las existentes. Posteriormente el proceso acaba.

Si no se indica *IMMEDIATE*, el proceso "dispatcher" espera hasta que todos sus usuarios se desconecten antes de finalizar.

CAMBIAR NUMERO "SHARED SERVER".

- Tras arrancar la instancia, puede modificarse el número mínimo de procesos "shared server" mediante la sentencia ALTER SYSTEM. Oracle elimina los servidores desocupados cuando hay más que el límite mínimo especificado.
- Si SHARED_SERVERS es 0, Oracle elimina todos los servidores actuales cuando se desocupan y no arranca ninguno hasta que se incrementa el valor de "SHARED_SERVERS".
- Por ejemplo, para fijar el mínimo de procesos "shared server" a cuatro:

```
ALTER SYSTEM SET SHARED_SERVERS = 4;
```

CAMBIAR NUMERO "SHARED SERVER".

- El número de "shared servers" en un momento determinado de la operación de la bd no puede fijarse -sólo el máximo y el mínimo son definibles-.
- Los "shared servers" se crean dinámicamente bajo el control automático de la arquitectura "shared server" de la instancia Oracle. Así, no se necesita ni se puede configurar el número de ellos en ningún momento.
- Conforme la carga se incrementa, por ejemplo al aumentar el número de consultas/DDLS/DMLs que los clientes ejecutan, la instancia Oracle incrementa el número de "shared servers" también -hasta el límite fijado por MAX_SHARED_SERVERS-. En el caso de que la carga descienda, los "shared servers" son eliminados automáticamente por la instancia.

FORZAR CONFIGURACION COMPARTIDA.

- Si está configurado "shared server" y llega una conexión cuando no hay "dispatcher" registrados, las peticiones pueden ser manejadas por un proceso servidor dedicado (configurado en *listener.ora*). Si se desea que un cierto cliente use siempre un "dispatcher", debe indicarse (*server=shared*) en el descriptor de conexión. Si no hay un "dispatcher" disponible, se rechaza la conexión del cliente.

```
sales=  
(DESCRIPTION=  
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)  
    (PORT=1521))  
(CONNECT_DATA=  
    (SERVICE_NAME=sales.us.acme.com)  
    (SERVER=shared)))
```

FORZAR CONFIGURACION DEDICADA.

- En ciertas ocasiones, interesa que un cliente se conecte a una instancia usando un servidor dedicado -como al ejecutar un trabajo "batch", o al usar Recovery Manager para copiar, recuperar o restaurar una bd-.

Debe hacerse una conexión mediante un servicio de red configurado como servidor dedicado:

- Bien el servicio contiene la clausula *SERVER=DEDICATED*.

sales=

(DESCRIPTION=

(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)

(PORT=1521))

(CONNECT_DATA=(SERVICE_NAME=sales.us.acme.com)

(SERVER=dedicated)))

FORZAR CONFIGURACION DEDICADA.

- Bien el fichero *sqlnet.ora* del cliente debe contener la variable `USE_DEDICATED_SERVER=on` -añade (*server=dedicated*) a la sección `CONNECT_DATA` del descriptor que el cliente use, el valor es sobrescrito-.

FORZAR CONFIGURACION.

- La instancia Oracle sólo proporciona la forma de limitar el número total de sesiones que pueden establecerse, como un conjunto, pero no deja limitar de forma específica sesiones dedicadas o sesiones compartidas como tales.
- Para limitar el número total de sesiones que pueden establecerse, puede usarse el parámetro SESSIONS en el fichero de parámetros de inicialización.

VISTAS.

- *V\$DISPATCHER* ... Información sobre los procesos "dispatcher" (nombre, estado, estadísticas,...).
- *V\$DISPATCHER_RATE* ... Estadísticas para "dispatcher".
- *V\$QUEUE* ... Información sobre las colas de mensaje de los servidores compartidos.
- *V\$SHARED_SERVER* ... Información de procesos "shared-server".
- *V\$CIRCUIT* ... Circuitos virtuales -conexiones de usuario a la bd a través de "dispatchers" y servidores.
- *V\$SHARED_SERVER_MONITOR* ... Información para ajuste de "shared server".
- *V\$SGA*
- *V\$SGASTAT* ... Información estadística de la SGA.
- *V\$SHARED_POOL_RESERVED*

TEMA 8.

ARCHIVADO DE BASE DE DATOS.

TEMA 8.

ARCHIVADO DE BASE DE DATOS.

- "REDO LOG" ARCHIVADO.
- MODO "NOARCHIVELOG".
- MODO "ARCHIVELOG".
- INFORMACION DE ARCHIVADO.
- MODO DE ARCHIVADO INICIAL.
- MODO ARCHIVELOG. ARCHIVADO AUTOMATICO.
- MODO ARCHIVELOG. ARCHIVADO MANUAL.

TEMA 8.

ARCHIVADO DE BASE DE DATOS.

- MODO ARCHIVELOG.DESTINO DE ARCHIVADO.
- MODO ARCHIVELOG. FALLOS DESTINO ARCHIVADO.
- MODO ARCHIVELOG. CONTROL DE PROCESOS ARCn.
- CAMBIO A MODO "ARCHIVELOG".
- INFORMACION PROCESO ARCHIVADO.
- CAMBIO A MODO "NOARCHIVELOG".
- VISTAS.

“REDO LOG” ARCHIVADO.

- Se llama “redo log” archivado al conjunto de ficheros de “redo log” archivados, que son copia de ficheros pertenecientes a un grupo de “redo log” -escritos al completo y en línea- de la base de datos.
- Al proceso de copia se le denomina proceso de archivado. Oracle arranca tantos procesos “archiver” (ARCn) como sean necesarios para lograr que el proceso sea correcto.
- La copia incluye las entradas de “redo” del fichero en línea y mantiene el número de secuencia único de “log” del grupo.

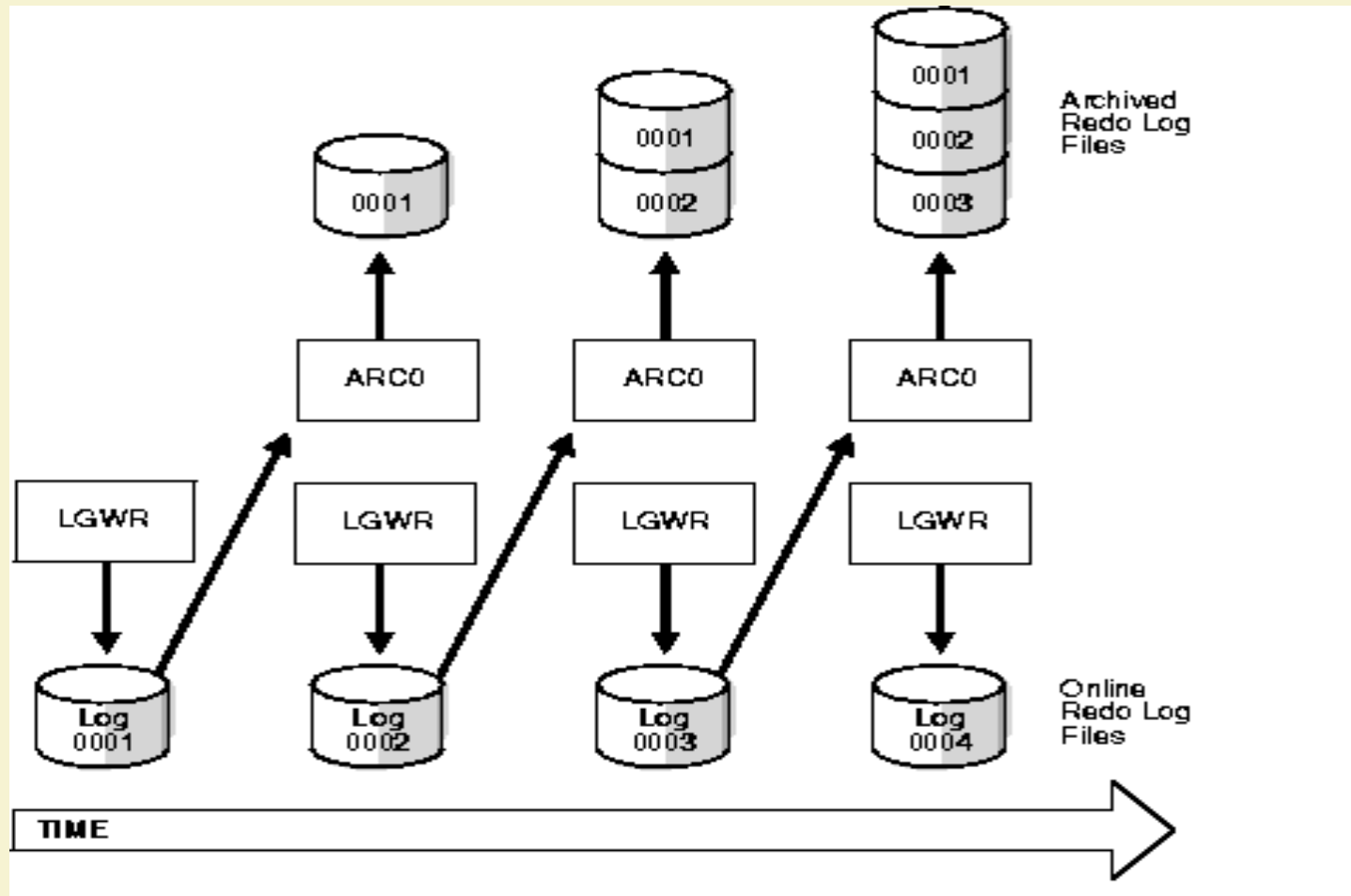
MODO "NOARCHIVELOG".

- Modo "noarchivelog":
 - Está desactivado el archivado de los "redo log" en línea. Cuando se completa el llenado de un grupo y se produce el "log switch", este se reutiliza por el LGWR.
 - En este modo la bd está protegida contra el fallo de instancia; pero si se produce un fallo en disco o en fichero, sólo puede recuperarse hasta la copia completa más reciente.
 - Deben hacerse periódica y frecuentemente copias completas de la bd.

MODO "ARCHIVELOG".

- Modo "archivelog":
 - Se archivan los "redo log" en línea. Cuando se completa el llenado de un grupo, este NO es reutilizado por el LGWR hasta que se archiva.
 - Pueden usarse copias realizadas mientras la base de datos está abierta y en funcionamiento.
 - Permite recuperar todas las transacciones validadas ("commit") en caso de fallo en disco o de sistema operativo.
 - Es aconsejable en el caso de no poder permitir la pérdida de datos producidas por los anteriores tipos de fallos.

MODO "ARCHIVELOG".



INFORMACION DE ARCHIVADO.

- La orden de SQL*Plus *ARCHIVE LOG LIST* se usa para mostrar información de archivado para una determinada instancia.
 - Modo de operación: ARCHIVELOG o NOARCHIVELOG.
 - Estado de archivado automático.
 - Destino de archivado.
 - Número de secuencia para el más antiguo “redo log” en línea lleno.
 - Idem para el próximo a archivar.
 - Idem para el actual.

MODO DE ARCHIVADO INICIAL.

- Indicado al crear la base de datos mediante la sentencia *CREATE DATABASE*. Por defecto es *NOARCHIVELOG*.

```
CREATE DATABASE [database]
{USER SYS IDENTIFIED BY ... | USER SYSTEM IDENTIFIED BY ...
| CONTROLFILE REUSE
| LOGFILE [GROUP integer] redo_log_file_spec
  [, [GROUP integer] redo_log_file_spec]...
| MAXLOGFILES integer | MAXLOGMEMBERS integer
| MAXLOGHISTORY integer | MAXDATAFILES integer
| MAXINSTANCES integer | { ARCHIVELOG | NOARCHIVELOG }
| FORCE LOGGING | CHARACTER SET charset | NATIONAL
  CHARACTER SET charset
| DATAFILE datafile_tempfile_spec [, datafile_tempfile_spec]...
| default_temp_tablespace | undo_tablespace_clause
| set_time_zone_clause }...;
```

CAMBIO A MODO "ARCHIVELOG".

- Se emplea la sentencia *ALTER DATABASE* y las cláusulas *NOARCHIVELOG* y *ARCHIVELOG* (sólo cuando la bd está montada pero no abierta y Real Application Clusters -RAC- deshabilitado).
- La cláusula *ARCHIVELOG* sólo puede usarse tras parar la instancia normalmente o de forma inmediata sin errores.

CAMBIO A MODO "ARCHIVELOG".

- Parar la instancia de base de datos:
 - » *shutdown*
- Realizar una copia de seguridad de la bd.
- Modificar el fichero de parámetros de inicialización.
- Arrancar una instancia y montar la bd:
 - » *startup mount*
- Modificar el modo de archivado y abrir la bd:
 - » *alter database archivelog;*
 - » *alter database open;*
- Comprobar el modo (*archive log list;*), archivar los "redo log" (*archive log all;*) y parar la instancia:
 - » *shutdown immediate*
- Realizar una copia de seguridad de la bd.

CAMBIO A MODO "ARCHIVELOG".

- El cambio de modo actualiza el fichero de control. Tras cambiarlo, debe realizarse una copia todos los ficheros de la base de datos.
- Cualquier copia anterior es desechable pues se realizó en modo *NOARCHIVELOG*.

MODULO ARCHIVELOG. ARCHIVADO AUTOMATICO.

- Al activar el archivado automático, Oracle guarda de forma automática la copia del fichero de "redo log" al llenarse.
- Activación tras el arranque de la instancia.
 - Necesario estar conectado AS SYSDBA o tener el privilegio ALTER SYSTEM.
 - Debe usarse la sentencia:
 - » *alter system archive log start;*
- Activación previa al arranque de la instancia.
 - Debe incluirse en el fichero de parámetros:
 - » *LOG_ARCHIVE_START=TRUE*

MODULO ARCHIVELOG. ARCHIVADO AUTOMATICO.

- El archivado automático puede desactivarse, a partir de ese momento debe archivarse de forma manual. Si se da el caso de que todos los grupos de "redo log" están completos y no se han archivado, LGWR no puede trabajar y la operación de la bd se suspende hasta realizar el archivado.
- Desactivación tras el arranque de la instancia.
 - Necesario estar conectado AS SYSDBA o tener el privilegio ALTER SYSTEM. Debe usarse la sentencia:
» *alter system archive log stop;*
- Desactivación previa al arranque de la instancia.
 - Debe incluirse en el fichero de parámetros:
» *LOG_ARCHIVE_START=FALSE*

MODO ARCHIVELOG. ARCHIVADO MANUAL.

- Si no está activo el archivado automático, debe archivarse de forma manual.
- Es necesario estar conectado AS SYSDBA o tener el privilegio ALTER SYSTEM. La sentencia:

» *alter system archive log all;*

guarda todos los ficheros aún no archivados.

- Mediante *alter system archive log current;* se fuerza a cambiar de "log" y archivar el que se está usando así como el resto de no archivados (util para copias "online").

MODO ARCHIVELOG. DESTINO DE ARCHIVADO.

- Puede ser único o múltiple, en este caso se guardan en más de una localización.
- El parámetro de inicialización *LOG_ARCHIVE_DEST_n* -siendo n un entero entre 1 y 10- permite especificar hasta diez destinos de archivado. El caso más común es el archivado local -en la propia maquina-, lo que se indica:

```
LOG_ARCHIVE_DEST_1 =  
  'LOCATION=/u06/oradata/CURSOxx'
```

- Es incompatible con el parámetro *LOG_ARCHIVE_DEST*.

MODO ARCHIVELOG. DESTINO DE ARCHIVADO.

- Para especificar el destino de archivado debe pararse la instancia de base de datos -shutdown- e:
 - Indicar en fichero de parámetros destinos de archivado.
 - » `LOG_ARCHIVE_DEST_1 = 'LOCATION = /u06/oradata/CURSOxx'`
 - » `LOG_ARCHIVE_DEST_2 = 'LOCATION = /u07/oradata/CURSOxx' ...`
 - Indicar el valor del parámetro `LOG_ARCHIVE_FORMAT`, usando `%s` para incluir el numero de secuencia de log como parte del nombre y `%t` par incluir el número de "thread". Si se emplean mayúsculas (`%S` and `%T`) el nombre se completa con ceros a la izquierda. Por ejemplo, `LOG_ARCHIVE_FORMAT = "LOG%s_%t.ARC"`

MODO ARCHIVELOG. DESTINO DE ARCHIVADO.

- La ruta completa de localización se forma concatenando ambas variables. Así, en el ejemplo anterior se generan ficheros de la forma:

/u06/oradata/CURSOxx/LOGyyy_zzz.ARC

/u07/oradata/CURSOxx/LOGyyy_zzz.ARC

- *Cada destino tiene una serie de características que determinan su estado (ver vista v\$ARCHIVE_DEST):*
 - Valid/Invalid ... si el destino se ha indicado y es válido
 - Enabled/Disabled ... estado de disponibilidad y puede ser usado.
 - Active/Inactive ... si hubo un problema accediendo al destino

MODO ARCHIVELOG. DESTINO DE ARCHIVADO.

- El parámetro *LOG_ARCHIVE_DEST_STATE_n* (dinámico) indica el estado del destino de localización:
 - *Enabled* ... indica un destino válido y que puede usarse. Es el valor por defecto.
 - *Defer* ... localización temporalmente deshabilitada.
 - *Alternate* ... destino no activo, pero que puede llegar a serlo si los accesos a otro destino fallan.

MODO ARCHIVELOG. FALLOS DESTINO ARCHIVADO.

- Con *LOG_ARCHIVE_MIN_SUCCEED_DEST=n* (dinámico) se determina el mínimo número de destinos a los que debe escribirse satisfactoriamente antes de poder usar el "redo log" en línea. Su valor por defecto es 1.
- Con *LOG_ARCHIVE_DEST_n* (dinámico) puede indicarse si un destino tiene como atributo *MANDATORY* -el archivado debe tener éxito antes de poder reutilizar el "redo log"- u *OPTIONAL* (por defecto) -el archivado no se requiere para poder usar el "redo log"-.

MODO ARCHIVELOG. FALLOS DESTINO ARCHIVADO.

- Con `LOG_ARCHIVE_DEST_n` puede especificarse el mínimo número de segundos antes de que el proceso ARCn intente acceder a un destino que ha fallado (opción REOPEN). Si el destino es del tipo MANDATORY, esta opción reduce el riesgo de se produzca un “shutdown” debido a falta de “redo log” en línea.
- El valor por defecto es de 300 segundos. Con un valor de cero, ARCn no intentará reescribir tras un fallo.

MODO ARCHIVELOG. FALLOS DESTINO ARCHIVADO.

- Si se indica REOPEN para un destino OPTIONAL, Oracle puede sobreescribir "redo" en línea si hay error. Si se indica REOPEN para un destino MANDATORY, Oracle stalls la bd cuando no puede archivar correctamente. En esta situación, debe:
 - Archivar manualmente al destino erróneo.
 - Cambiar el destino (DEFER, OPTIONAL, ...).
 - Eliminar el destino.

MODULO ARCHIVELOG. CONTROL DE PROCESOS ARCn.

- Los procesos de archivado (ARCn) son creados conforme se necesitan.
- Puede indicarse el número de procesos creados en el arranque de la instancia, añadiendo en el fichero de parámetros:

```
LOG_ARCHIVE_MAX_PROCESSES=<numero>;
```

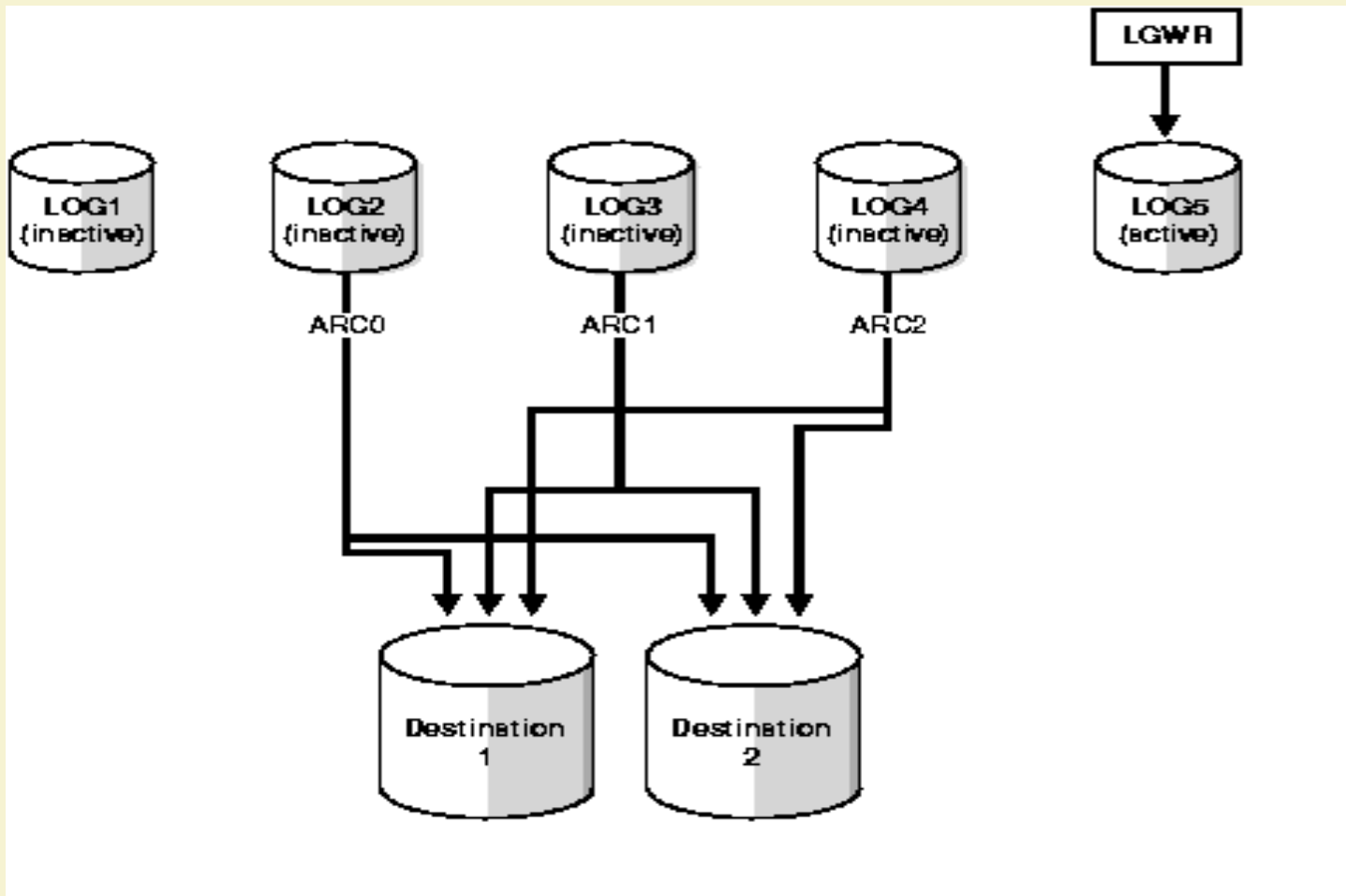
- No pudiendose crear más de los indicados.
- Este número también puede modificarse dinámicamente con la sentencia:

```
ALTER SYSTEM SET  
LOG_ARCHIVE_MAX_PROCESSES=<numero>;
```

MODO ARCHIVELOG. CONTROL DE PROCESOS ARCn.

- Crear múltiples procesos es útil cuando:
 - Se utilizan más de dos “redo log” en línea.
 - Se archiva a más de un destino.
- Se previene así el cuello de botella que sucede cuando LGWR permuta entre los “redo log” en línea más rápidamente que lo que tarda el proceso ARCn en escribir “logs” inactivos a sus destinos.
- Cada proceso ARCn trabaja sólo con un “log” inactivo cada vez, pero debe escribir a cada uno de los destinos indicados.

MODO ARCHIVELOG. CONTROL DE PROCESOS ARCn.



INFORMACION PROCESO ARCHIVADO.

- Se controla con el parámetro -modificable con *ALTER SYSTEM- LOG_ARCHIVE_TRACE*. Valores válidos son:
 - 0: Disable archivelog tracing (valor por defecto)
 - 1: Track archival of redo log file
 - 2: Track archival status of each archivelog destination
 - 4: Track archival operational phase
 - 8: Track archivelog destination activity
 - 16: Track detailed archivelog destination activity
 - 32: Track archivelog destination parameter modifications
 - 64: Track ARCn process state activity
 - 128: Track FAL (fetch archived log) server related activities

INFORMACION PROCESO ARCHIVADO.

- Pueden combinarse sumandolos entre ellos (por ejemplo, el valor 3 hace referencia a 1 y a 2).
- Al fijar el valor por defecto, cero, Oracle continuará generando entradas en el fichero de "alert" y trazas en respuesta a condiciones de error.

CAMBIO A MODO "NOARCHIVELOG".

- Parar la instancia de base de datos:
 - » *shutdown*
- Realizar una copia de seguridad de la bd.
- Modificar el fichero de parámetros de inicialización.
- Arrancar una instancia y montar la bd:
 - » *startup mount*
- Modificar el modo de archivado:
 - » *alter database noarchivelog;*
 - » *alter database open;*
- Comprobar el modo (*archive log list*) y parar la instancia:
 - » *shutdown immediate*
- Realizar una copia de seguridad de la bd.

VISTAS.

- *V\$DATABASE*. Consultar si la base de datos está en modo *ARCHIVELOG* o *NOARCHIVELOG*.
- *V\$ARCHIVED_LOG*. Muestra información histórica sobre archivado del fichero de control.
- *V\$ARCHIVE_DEST*. Información sobre destino de archivado.
- *V\$ARCHIVE_PROCESSES*. Información sobre el estado de los procesos de archivado.
- *V\$BACKUP_REDOLOG*. Información sobre "backups" de los "logs" archivados.
- *V\$LOG*. Muestra los grupo de "redo" en línea para la bd e indica cual necesita archivarse.
- *V\$LOG_HISTORY*. Información histórica de "log".

TEMA 9.

COPIAS DE SEGURIDAD.

TEMA 9.

COPIAS DE SEGURIDAD.

- COPIAS FISICAS/LOGICAS.
- COPIAS COMPLETAS/PARCIALES.
- COPIA. CONSIDERACIONES.
- ESTRATEGIA MANUAL DE COPIA.
- FICHEROS A COPIAR.
- COPIA.
 - COMPLETA BD.
 - TBSP FUERA LINEA.
 - TBSP LECTURA/ESCRITURA EN LINEA. PROBLEMAS.
 - TBSP SOLO LECTURA EN LINEA.
 - TBSP UNDO.
 - MODO "SUSPEND".
 - FICHERO CONTROL.

TEMA 9.

COPIAS DE SEGURIDAD.

- “REDO” ARCHIVADO.
- OTROS FICHEROS.
- DBVERIFY: OFFLINE DB VERIFICATION UTILITY.
 - VALIDACION FICHEROS.
 - VALIDACION SEGMENTOS.
- UTILIDADES EXPORT/IMPORT.
- “TBSP” TRANSPORTABLES.
 - LIMITACIONES.
 - COMPATIBILIDAD.
 - VARIABLES.
 - PROCESO.
- VISTAS.

COPIA Y RECUPERACION. METODO MANUAL vs. RMAN.

- Oracle recomienda usar RMAN como base de la estrategia de copia y recuperación, pero los métodos manuales también son útiles.
- Diferentes circunstancias pueden justificar el uso de los métodos manuales:
 - Migración de bd a una versión actualizada pero conservando aún los "script" de copia actuales.
 - Coexistencia de bbdd tipo Oracle7 junto a otras más recientes, y la necesidad de coherencia en el manejo de todas ellas (RMAN sólo soporta bd versión 8.0 o superior).
 - Pérdida de las copias realizadas con RMAN, y necesidad de restaurar copias manuales y recuperar con la orden SQL*Plus *RECOVER*.

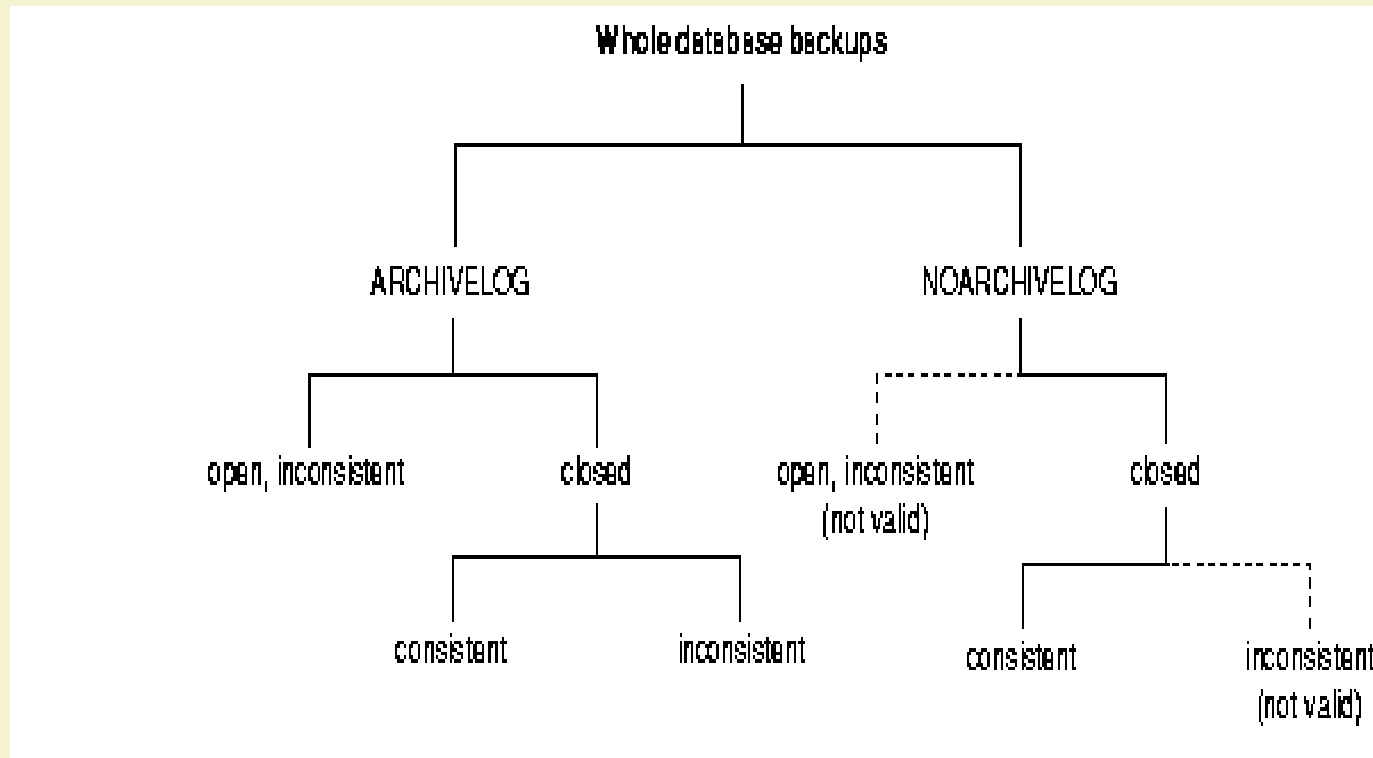
COPIAS FISICAS/LOGICAS.

- Debe permitir, en caso de pérdida de los datos originales, recuperarlos.
- Existen diferentes criterios de clasificación:
 - Físicas. Copias de los ficheros que integran la bd, ficheros de control y ficheros "archive log" hechas con utilidades del gestor de bd -*Recovery Manager, RMAN*- o del s.o..
 - Lógicas. Exportación a fichero binario mediante la utilidad *EXPORT*.

Ambas son complementarias.

COPIAS COMPLETAS/PARCIALES.

- **Completas.** Incluyen ficheros de control y de datos. Es independiente del modo de operación de la bd.



COPIAS COMPLETAS/PARCIALES.

- Una copia completa puede ser:
 - Consistente. Puede abrirse con la opción RESETLOGS sin realizar "media recovery", o sea, sin aplicar "redo" a los ficheros de datos. Todos los ficheros de datos deben:
 - Tener el mismo "system change number" (SCN) en sus cabeceras, a menos que los tbsps correspondientes estén en sólo lectura u "offline normal".
 - No contienen cambios tras el SCN.
 - Concuerdan con la información de "checkpoint" almacenada en el fichero de control.

*Nota: Un "**system change number**" (SCN) es una marca de tiempo, "timestamp", interna para una versión validada de la bd. La bd emplea el valor de reloj SCN para garantizar la consistencia de las transacciones. Por ejemplo, cuando un usuario valida una transacción, Oracle graba un SCN para esta validación en el "redo log" en línea.*

Oracle también emplea SCNs para coordinar transacciones distribuidas entre distintas bdd

COPIAS COMPLETAS/PARCIALES.

Sólo pueden efectuarse copias consistentes tras un "shutdown" limpio de la bd, esta no debe abrirse hasta finalizar la misma.

Una copia completa y consistente es la única opción válida para bd en modo NOARCHIVELOG.

- *Inconsistente*. Copia donde algun fichero contiene cambios hechos después del "checkpoint" de los ficheros. Necesita recuperación antes de ser consistente.

Normalmente se hacen con la bd abierta mientras se realiza la copia. Pero también si la copia se hace sobre una bd cerrada justo después de caer una instancia o de cerrar la bd con SHUTDOWN ABORT.

Sólo son válidos si la bd está en modo archiveolog.

COPIAS COMPLETAS/PARCIALES.

- **Parciales.**

- Copias de los ficheros de un "tablespace". Sólo son válidos si la bd está en modo archivelog (en modo noarchivelog sólo cuando está en modo sólo lectura u "offline normal").

- Pueden hacerse con RMAN o mediante el s.o..

- Copia de un fichero de datos. Válido en modo archivelog (en modo noarchivelog cuando está en modo sólo lectura u "offline normal", o cuando todo fichero de un tbsp. se copia). Pueden hacerse con RMAN o mediante el s.o..

- Copia de fichero de control. Mediante RMAN, o la sentencia "alter database backup controlfile" -a binario-.

- Copias de "redo log" archivados. Fundamentales para recuperar una copia inconsistente. Deben copiarse todos los producidos durante una copia inconsistente.

COPIAS COMPLETAS/PARCIALES.

- Durante las copias realizadas en línea pueden producirse inconsistencias en los datos que se intenta copiar (por ejemplo, la lectura de bloques fracturados -con datos antiguos y nuevos al estar modificándose-).
- Para evitarlo, el tbsp. debe ponerse en modo "backup" con la sentencia "*alter tablespace begin backup*" (no puede indicarse para un tbsp de sólo lectura ni un tbsp temporal gestionado localmente). Mientras dura la copia no puede ponerse fuera de línea, parar la instancia o comenzar otra copia del tbsp.
- Mediante "*alter tablespace end backup*" se indica que la copia del tbsp. ha acabado. Se recomienda usarla tan pronto como sea posible, pues si sucede un fallo de instancia o "shutdown abort", Oracle piensa que se necesita "media recovery" en el siguiente arranque (quizá con "redo log" archivado).

COPIA. CONSIDERACIONES.

- Anotar la configuración "hw" y "sw" del servidor.
- Conjunto de Redundancia.

Está integrado por los ficheros necesarios para recuperar la bd: *Copia del fichero de control y de los ficheros de datos, "redo log" archivado, duplicado de los "redo log" en línea y del fichero de control actual, y ficheros de configuración.*

El conjunto de redundancia debe estar en discos distintos de aquellos que contienen ficheros de datos, "redo" en línea y ficheros de control. También se aconseja mantener distintas copias del conjunto.

- Guardar múltiples copias de los ficheros de control, "redo log" en línea y "redo log" archivado.

COPIA. CONSIDERACIONES.

- Copia en modo ARCHIVELOG.
 - Copiar la bd al completo tras su creación, o paso a modo ARCHIVELOG.
 - Realizar copias de los tbsp con la bd abierta o cerrada (no es necesario cerrar la bd). En particular de aquellos de uso intensivo deben hacerse copias frecuentes (para reducir el tiempo de recuperación).
 - Copiar el fichero de control cada vez que se realice un cambio estructural en la bd (*"ALTER DATABASE BACKUP CONTROLFILE"*).
 - Copiar los "redo log" archivados frecuentemente. Es conveniente realizar copias a cinta, además de a disco.

COPIA. CONSIDERACIONES.

- Realizar copias de manera frecuente y regular (más frecuencia a mayor número de operaciones DML).
- Realizar copias antes y después de cambios en la estructura de la bd (creado/borrado de tbsp, renombrado de fichero de datos, ...).
- Realizar copia después de operaciones con la opción *UNRECOVERABLE* (create table, create index) -sustituida por la opción LOGGING/NOLOGGING-.
- Realizar copia completa después de abrir la bd con la opción *RESETLOGS*.
- Guardar las copias antiguas de la bd.

COPIA. CONSIDERACIONES.

- Realizar exportaciones de la bd (como complemento a las copias).
- Evitar la copia de "redo log" en línea.
 - La mejor forma de protegerlos es tener múltiples miembros en cada grupo, y en diferentes discos.
 - Si la bd está en modo ARCHIVELOG, el proceso ARC guarda los "redo" que se han llenado. Si la bd está en modo NOARCHIVELOG, el único tipo de copias válido es el realizado con la bd cerrada, consistente y completo; los ficheros en este tipo de copia son todos consistentes y no necesitan recuperación ni, por tanto, los "redo" en línea.
 - Pueden restaurarse, por error, copias de los "redo log" en línea y corromper la bd.

ESTRATEGIA MANUAL DE COPIA.

- Se realizan copias periódicas de los ficheros de datos y "redo" archivado usando órdenes de s.o..

Objeto	Actuación
Ficheros de datos	<i>cp <nombre_fichero> <nombre_fichero.fecha></i>
F."redo" archivado	<i>cp <nombre_fichero> <nombre_fichero.fecha></i>
F. control	<i>alter database backup controlfile to <nombre_fichero></i>
F. parámetros	<i>create pfile= <init.ora.fecha> from spfile;</i>
F. red	<i>cp tnsnames.ora tnsnames.ora.fecha</i>

ESTRATEGIA MANUAL DE COPIA.

- Método básico de copia de la bd al completo:
 - Identificar ficheros de datos, de control y de "redo" archivados a copiar (no deben copiarse los "redo" en línea).
 - Copiar los ficheros de datos y de "redo" identificados en el paso anterior.
 - Emplear una sentencia sql para copiar el fichero de control.
 - Copiar los ficheros de configuración.

FICHEROS A COPIAR.

- Identificar los ficheros a copiar.
 - Se consultan las vistas *V\$DATAFILE* -ficheros de datos-, *V\$TABLESPACES* -tbsp-, *V\$LOGFILE* -"redo" en línea- y *V\$CONTROLFILE* -f.control-. Al hacer copia con "*ALTER DATABASE BACKUP CONTROLFILE TO <fichero>;*" debe guardarse información sobre los f.datos y f.redo en línea existentes (la estructura puede cambiar en el momento de recuperar).
- Determinar el estado de copia (vista *V\$BACKUP*).
 - Muestra el estado de copia de los f.datos en línea -el valor *ACTIVE* de la columna "status" indica en copia- (útil con bd abierta y con f.control no recuperado ni nuevo -en este caso, contendrían información desactualizada-).

COPIA COMPLETA BD.

- Es posible realizarla tras cerrar la bd, "*shutdown*", con las opciones: *Normal*, *immediate* o *transactional*.
- Puede realizarse en modo *archivelog* (la copia puede ser inconsistente) o *noarchivelog* (copia consistente).
- Puede restaurarse sin hacer recuperación (en modo *archivelog* puede recuperarse posteriormente para llevar la bd a un momento más actual).
- En modo *archivelog* se recomienda hacer copia del f.control con "*ALTER DATABASE BACKUP CONTROLFILE TO <fichero>;*"

COPIA COMPLETA BD. PROCESO.

- Cerrar la bd.

```
SQL> SHUTDOWN NORMAL / IMMEDIATE / TRANSACTIONAL
```

No debe realizarse copia de la bd si ha sido parada de forma anormal ("abort" o caída debido a un fallo).

- Copiar los ficheros apropiados (datos, control, inicialización, ...). Buscar todos los ficheros .ora comenzando por el \$ORACLE_HOME.
- Rearrancar la bd.

```
SQL> STARTUP
```

COPIA TBSP FUERA LINEA.

- Puede realizarse la copia de todos o parte de los f.datos que integran un tbsp cuando está fuera de línea (para ponerlo fuera de línea debe tenerse el privilegio *DBA* o *MANAGE TABLESPACE*).
- No puede ponerse fuera de línea el tbsp *SYSTEM* o un tbsp con segmentos de "rollback" activos.
- Si el tbsp fuera de línea contiene índices al intentar acceder a estos se generan errores.

COPIA TBSP FUERA LINEA. PROCESO.

- Identificar los ficheros de datos correspondientes a los tbsp (*DBA_DATA_FILES*).
- Poner el tbsp fuera de línea con prioridad normal (*ALTER TABLESPACE <nombre_tbsp> OFFLINE NORMAL;*).
- Realizar copia de los f.datos (*cp*).
- Poner el tbsp en línea (*ALTER TABLESPACE <nombre_tbsp> ONLINE;*).
- Forzar el archivado de "redo", de forma que aquel necesario para recuperar la copia del tbsp esté disponible (*ALTER SYSTEM ARCHIVE LOG CURRENT;*).

COPIA TBSP LECTURA/ESCRITURA EN LINEA. PROCESO.

- Identificar los f.datos que pertenecen al tbsp.
- Indicar el comienzo de copia:
"alter tablespace <nombre_tbsp> begin backup;"
- Realizar la copia (cp).
- Indicar fin de copia:
"alter tablespace <nombre_tbsp> end backup;"
- Forzar el archivado de "redo", de forma que aquel necesario para recuperar la copia esté disponible:
"alter system archive log current;"

COPIA TBSP LECTURA/ESCRITURA EN LINEA.

Si el proceso de copia involucra a varios tbsp, pueden copiarse de forma paralela o en serie (en ambos casos se fuerza el archivado al final del proceso, cuando sobre todos los tbsp se haya hecho el "end backup").

- Paralela.

Se ponen simultáneamente todos los tbsp en modo "begin backup" -implica un gran crecimiento de "redo" en línea al actualizarlos-. Se realiza la copia y se hace "end backup".

- Serie (Opción recomendada).

Repetir el proceso "begin backup"-copia-"end backup" para cada tbsp. Minimiza el tiempo "begin"- "end" para un tbsp.

PROBLEMAS EN LA COPIA.

- Una copia de tbsp puede fallar y no completarse cuando:
 - La copia ha terminado pero no se indica su fin ("end backup").
 - Falla la instancia o se hace un "shutdown abort".
 - En caso de un "crash recovery" si un fichero está en modo copia y se intenta abrir, Oracle supone que es una copia a recuperar y no se abre hasta la recuperación o hasta abandonar el modo copia (se generan los ORA-01113 y ORA-01110).

PROBLEMAS EN LA COPIA.

- Abandonar el modo de copia.
 - Montar la bd.
 - Consultar en `V$BACKUP` tbsp en copia (`status='ACTIVE'`).
 - Ejecutar `alter database end backup;` (sólo si la bd está montada y no se ha recuperado ningun fichero implicado).
 - Abrir la bd.
- Recuperación (caso de no estar seguro de haber restaurado copia).
 - Montar la bd.
 - Recuperar la bd (`RECOVER DATABASE`) y comprobar en `V$BACKUP` que no hay tbsp en copia (`status='ACTIVE'`).
 - Abrir la bd.

COPIA TBSP SOLO LECTURA EN LINEA. PROCESO.

- No es necesario poner el tbsp en modo copia, simplemente copiar los f.datos correspondientes. Si el conjunto de tbsp es autocontenido puede además realizarse la exportación del tbsp (tbsp transportable).
 - Consultar *DBA_TABLESPACES* para determinar qué tbsp están en modo sólo lectura.
 - Identificar los ficheros asociados a los tbsp consultando *DBA_DATA_FILES*.
 - Copiar los ficheros correspondientes.
 - Al restaurar la copia, debe ponerse fuera de línea el tbsp, copiar los ficheros y ponerlo en línea.
 - La copia es válida incluso si se cambia a lectura/escritura tras la copia -necesitará recuperación-.

COPIA TBSP UNDO.

- El procedimiento es el mismo usado para la copia de tbsp lectura/escritura. Al ser el espacio de "undo" importante para la recuperación y la consistencia en lectura, debe copiarse frecuentemente.
- Si los f.datos que integran el tbsp de "undo" se perdieran mientras la bd está abierta, y no se dispusiera de copia, podrían recibirse mensajes de error al consultar objetos que contuvieran cambios no validados. Además si falla la instancia, no se podrían retroceder las transacciones no validadas a su valor original.

COPIA MODO "SUSPEND".

- Existen herramientas que permiten hacer copias "espejo" o duplicados exactos y desvincular la copia del original de forma que puedan usarse independientemente.
- Con la funcionalidad *SUSPEND/RESUME* puede suspenderse la entrada/salida de la bd, desvincular la copia "espejo" y hacer una copia de la misma (en este estado no puede ejecutarse *SHUTDOWN IMMEDIATE* o *NORMAL*; ejecutar *SHUTDOWN ABORT* reactiva la bd).
- La sentencia *ALTER SYSTEM SUSPEND* permite "suspender" la bd, todas las operaciones E/S se completan pero cualquier nueva operación se guarda en cola. Se recomienda que se preceda con una sentencia "begin backup" (inicio de copia).
- Tras la copia se ejecuta la sentencia *ALTER SYSTEM RESUME*.

COPIA MODO "SUSPEND".

- Poner los tbsp en modo copia:
ALTER TABLESPACE <nombre> BEGIN BACKUP;
- Suspende la bd:
ALTER SYSTEM SUSPEND;
- Comprobar el cambio de estado (consultar *V\$INSTANCE*, campo *DATABASE_STATUS*).
- Realizar la copia.
- Finalizar la suspensión de la bd:
ALTER SYSTEM RESUME;
- Comprobar que la bd está activa (consultar *V\$INSTANCE*, campo *DATABASE_STATUS*).
- Poner los tbsp en modo no copia:
ALTER TABLESPACE users END BACKUP;
- Copiar el f.control y "redo" en línea y archivados.

COPIA FICHERO CONTROL.

- En una bd en modo archivelog debe copiarse tras realizar un cambio estructural. Se requiere el privilegio *ALTER DATABASE*.
- Puede copiarse a un fichero binario (preferible al contener información sobre historia de "redo" archivado, ...) o a un fichero de traza.

- Fichero binario.

– *ALTER DATABASE BACKUP CONTROLFILE TO
'/oracle/copia/cf.bak' REUSE;*

La opción REUSE permite sobreescribir un f.control que ya exista.

COPIA FICHERO CONTROL.

- Fichero de traza.
 - Se generan sentencias SQL de creación del fichero de control en un fichero de traza (arrancan la bd, recrean el f.control, y recuperan y abren la bd).
 - Debe montarse o abrirse la bd y ejecutar la sentencia:

ALTER DATABASE BACKUP CONTROLFILE TO TRACE;

- Si no se indica *RESETLOGS* ni *NORESETLOGS* en la sentencia, la salida es un fichero de traza que contiene la sentencia *CREATE CONTROLFILE ... NORESETLOGS.*
- En ambos casos (copia a fichero binario o de traza), las entradas para los ficheros "TEMP" no son incluidas.

COPIA "REDO" ARCHIVADO.

- Los ficheros de "redo" archivado generados pueden copiarse a cinta u otro dispositivo y, de esta forma, ahorrar espacio donde este localizada la bd.
- Para realizar la copia, se averiguan los ficheros de "redo" archivado que la bd ha generado:

```
SELECT THREAD#,SEQUENCE#,NAME  
FROM V$ARCHIVED_LOG;
```

- Se realiza una copia de cada número de secuencia de "log" (en caso de guardar múltiples copias, sólo es necesario una por número de secuencia).

COPIA OTROS FICHEROS.

- Como norma general, cada vez que se modifiquen deben copiarse además el fichero de parámetros de inicialización, el de interconexión, y el de contraseñas.
- La forma más sencilla de localizarlos es realizar una búsqueda automática desde \$ORACLE_HOME de todos los ficheros acabados en ".ora" (orden find de s.o. UNIX).

DBVERIFY: DATABASE VERIFICATION UTILITY.

- Es una utilidad que hace un chequeo de integridad de la estructura de datos física de una base de datos fuera de línea.
- Puede usarse en ficheros de copia ("backup") y ficheros en línea (o unidades menores). Generalmente se intenta asegurar así que una copia de bd (o fichero) es válida antes de restaurarla, o como herramienta de diagnóstico cuando se encuentran problemas de corrupción de datos.
- Al ejecutarse sobre una bd fuera de línea, los chequeos son mucho más rápidos. Estos se limitan a bloques de datos -no a ficheros de control ni "redo logs".

DBVERIFY. VALIDACION FICHEROS.

- Se verifican uno o más bloques de disco pertenecientes a un fichero de datos y se realizan chequeos de página.

```
dbv FILE = <nombre_de_fichero>  
| { START = <dirección_bloque> | END =  
  <dirección_bloque> }  
| BLOCKSIZE = integer  
| LOGFILE = <nombre_de_fichero>  
| FEEDBACK = integer  
| HELP = { Y | N }  
| PARFILE = <nombre_de_fichero>
```

Por ejemplo, *dbv FILE=nomina.dbf FEEDBACK=100*

DBVERIFY. VALIDACION FICHEROS.

- Los parámetros tienen el siguiente significado:
 - *FILE* ... Nombre del fichero de bd a verificar.
 - *START* ... Dirección inicial del bloque Oracle a verificar (por defecto, el primero del fichero).
 - *END* ... Dirección del último bloque a verificar (por defecto el último en el fichero).
 - *BLOCKSIZE* ... Necesario si el fichero no tiene un tamaño de bloque de 2 KB. Si es así y no se indica se genera el error DBV-00103.

DBVERIFY. VALIDACION FICHEROS.

- *LOGFILE* ... Fichero donde se guarda información (por defecto el terminal).
- *FEEDBACK* ... Informe de progreso de la verificación enviado al terminal. Un punto (.) por cada n número de página verificadas. Si $n = 0$, no hay informe.
- *HELP* ... Ayuda en línea.
- *PARFILE* ... Fichero de parámetros.

DBVERIFY. VALIDACION FICHEROS.

```
/home/oracle> dbv file=/u02/.../.../users01.dbf feedback=100  
DBVERIFY: Release 9.2.0.1.0 - Production on Mon Mar 14 20:42:43 2005  
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
```

DBVERIFY - Verification starting : FILE = users01.dbf

.....

DBVERIFY - Verification complete

Total Pages Examined : 2048

Total Pages Processed (Data) : 0

Total Pages Failing (Data) : 0

Total Pages Processed (Index): 0

Total Pages Failing (Index): 0

Total Pages Processed (Other): 32

Total Pages Processed (Seg) : 0

Total Pages Failing (Seg) : 0

Total Pages Empty : 2016

Total Pages Marked Corrupt : 0

Total Pages Influx : 0

DBVERIFY. VALIDACION FICHEROS.

Pages = bloques

Total Pages Examined = número de bloques en el fichero

Total Pages Processed = número de bloques que se verificaron
(formatted blocks)

Total Pages Failing (Data) = número de bloques que fallaron la
rutina de chequeo de bloque de datos.

Total Pages Failing (Index) = idem para bloque de índices.

DBVERIFY. VALIDACION FICHEROS.

Total Pages Marked Corrupt = número de bloques para los que la cabecera de caché no es válida, haciendo que fuera imposible determinar el tipo de bloque.

Total Pages Influx = número de bloques que están siendo leídos y escritos en el mismo tiempo. Si la bd está abierta, se leen los bloques múltiples veces para generar una imagen consistente. Por esta misma razón, pueden haber bloques leídos y escritos a la vez (INFLUX) para los que no se puede obtener una imagen consistente.

DBVERIFY. VALIDACION SEGMENTOS.

- Permite indicar un segmento de tabla o índice a verificar.
- Requiere la conexión a la bd con privilegios de SYSDBA, e indicar el segmento concreto a validar.
- Durante la comprobación, el segmento es bloqueado. Si se trata de un segmento de índice, la tabla padre es bloqueada asimismo (algunos índices como los index-organized table -IOT-, donde se mantiene las filas de la tabla en un índice construido en la clave primaria, no tienen tabla padre).

DBVERIFY. VALIDACION SEGMENTOS.

```
dbv USERID = usuario/contraseña  
| SEGMENT_ID = tsn.segfile.segblock  
| LOGFILE = <nombre_fichero>  
| FEEDBACK = integer  
| HELP = { Y | N }  
| PARFILE = <nombre_fichero>
```

- Los parámetros tienen el siguiente significado:
 - *SEGMENT_ID* ... Segmento a verificar. Debe identificarse el número de tbsp -ts# de v\$tablespace- , segfile, y segblock -header_file y header_block de dba_segments-.
 - LOGFILE ... Fichero donde se guarda información (por defecto el terminal).

DBVERIFY. VALIDACION SEGMENTOS.

- *FEEDBACK* ... Informe de progreso de la verificación enviado al terminal. Un punto (.) por cada n número de página verificadas. Si $n = 0$, no hay informe.
- *HELP* ... Ayuda en línea.
- *PARFILE* ... Fichero de parámetros.

DBVERIFY. VALIDACION SEGMENTOS.

```
SQL> select a.name tablespace, a.ts# numero_tbsp, b.SEGMENT_NAME  
segmento, b.HEADER_FILE cabecera, b.HEADER_BLOCK bloque from  
v$tablespace a, dba_segments b where a.name=b.TABLESPACE_NAME  
and b.tablespace_name='SYSTEM' and b.owner='SCOTT' order by  
b.SEGMENT_NAME;
```

```
...  
SYSTEM                0  
DEPT          1      57341  
...
```

DBVERIFY. VALIDACION SEGMENTOS.

```
/home/oracle (CURSO01)> dbv userid=system/systxxxx  
segment_id=0.1.57341
```

DBVERIFY: Release 9.2.0.1.0 - Production on Fri Mar 18 08:42:38 2005

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

DBVERIFY - Verification starting : SEGMENT_ID = 0.1.57341

DBVERIFY - Verification complete

Total Pages Examined : 2

Total Pages Processed (Data) : 1

Total Pages Failing (Data) : 0

Total Pages Processed (Index): 0

Total Pages Failing (Index): 0

Total Pages Processed (Other): 0

Total Pages Processed (Seg) : 1

Total Pages Failing (Seg) : 0

Total Pages Empty : 0

Total Pages Marked Corrupt : 0

Total Pages Influx : 0

UTILIDADES EXPORT/IMPORT.

- Son usadas para:
 - Transferir objetos de datos entre bases de datos Oracle (independientemente de la plataformas en que residan).
 - Proporcionar copia de seguridad **lógica** para objetos de la base de datos, un espacio de almacenamiento o la base de datos al completo.
 - Migración entre versiones de base de datos (¡Peligro!).
- La copia lógica realizada con “export” se almacena en un fichero binario; este fichero sólo puede ser leído por la utilidad “import”. Son utilidades complementarias.

UTILIDAD EXPORT.

- Para usar la utilidad debe tenerse el privilegio *CREATE SESSION*.
- Para exportar objetos pertenecientes a otro esquema de usuario debe tenerse asignado el rol *EXP_FULL_DATABASE*.
- Como resultado se obtiene un fichero *.dmp* (fichero de exportación) y, opcionalmente, un fichero *.log* con el informe de incidencias.
- Es conveniente que todos los ficheros generados para usar en la exportación o como resultado de ella estén en un sistema de ficheros independiente (p.ej. */export/<nombre_bd>*).

UTILIDAD EXPORT.

- Aunque puede ejecutarse la utilidad e ir indicando en el “prompt” diferentes opciones, se recomienda usar un fichero de parámetros.
- Sintaxis:

exp HELP=Y Proporciona ayuda en línea.

exp Modo interactivo (sin modo directo).

exp PARFILE= <fichero_parametros>

UTILIDAD EXPORT.

Parámetro	Descripción
BUFFER	Tamaño "bufer" de datos (bytes). No tiene efecto al usar "direct path".
COMPRESS (Y/N)	Incluir todos los datos en una extensión (no para LOB).
CONSISTENT (Y/N)	Asegura la consistencia de los datos exportados cuando pueden estar siendo actualizados (p.ej. con la bd abierta). Incompatible con copias incrementales.
CONSTRAINTS (Y/N)	Exportar o no las restricciones sobre tablas.
DIRECT (Y/N)	Modo directo o no -convencional- de exportación.
FILE	Nombre del fichero de "export", por defecto "expdat.dmp".
FULL (Y/N)	Exportación de la base de datos completa o no.

UTILIDAD EXPORT.

Parámetro	Descripción
GRANTS (Y/N)	Exportar o no privilegios sobre objetos.
INDEXES (Y/N)	Exportar o no índices.
LOG	Indica fichero donde se guardarán los mensajes.
OWNER	Indica los usuarios de los que se realizara la copia.
ROWS (Y/N)	Exportar o no los datos de las tablas.
TABLES	Relación de tablas a exportar (modo tabla).
TABLESPACES	Lista de "tablespaces" a exportar.
TRANSPORT_TABLESPACE (Y/N)	Permite exportación de "tablespaces".

UTILIDAD EXPORT.

Parámetro	Descripción
TRIGGERS (Y/N)	Exportar o no disparadores.
USERID	Usuario y contraseña del usuario que realiza la exportación (¡Peligro!).

UTILIDAD IMPORT.

- Para usar la utilidad debe tenerse el privilegio *CREATE SESSION*. Por supuesto, deben tenerse los privilegios necesarios para crear o trabajar con los objetos a importar.
- Un usuario puede importar un fichero de "export" no creado por él. Si el fichero de exportación fue creado con el privilegio *EXP_FULL_DATABASE*, debe tenerse asignado el rol *IMP_FULL_DATABASE*.
- Como resultado se obtiene un fichero *.log* con el informe de incidencias.

UTILIDAD IMPORT.

- Aunque puede ejecutarse la utilidad e ir indicando en el “prompt” diferentes opciones, se recomienda usar un fichero de parámetros.
- Sintaxis:

imp HELP=Y *Proporciona ayuda en línea.*

imp *Modo interactivo.*

imp PARFILE= <fichero_parametros>

UTILIDAD IMPORT.

Parámetro	Descripción
BUFFER	Tamaño "bufer" de datos (bytes) usado en la transferencia.
COMMIT (Y/N)	Hacer "commit" o no después de cada inserción. Útil para evitar que los segmentos de "rollback" crezcan demasiado. Si ocurre un error y no hay clave única se producirán filas duplicadas.
CONSTRAINTS (Y/N)	Importar o no las restricciones sobre tablas.
DESTROY (Y/N)	Indica si los ficheros de datos existentes debe ser o no reutilizados.
FILE	Nombre del fichero de "export" a importar, por defecto <i>expdat.dmp</i> .
FROMUSER	Lista de esquemas a importar (usado normalmente con TOUSER).

UTILIDAD IMPORT.

Parámetro	Descripción
FULL (Y/N)	Importar o no el fichero completo.
GRANTS (Y/N)	Importar o no privilegios sobre objetos.
IGNORE (Y/N)	<p>Indica la forma de tratar los errores generados en la creación de objetos. Si <i>IGNORE=Y</i> se ignoran los errores de creación y se continua, en caso contrario se muestran.</p> <p>En el caso de tablas, si <i>IGNORE=Y</i> se importan los datos en las tablas existentes. Si <i>IGNORE=N</i> se genera un error y no se inserta.</p>
INDEXES (Y/N)	Importar o no índices.
INDEXFILE	Fichero de descarga para sentencias de creación de índices, no son creados.

UTILIDAD IMPORT.

Parámetro	Descripción
LOG	Indica fichero donde se guardaran los mensajes.
ROWS (Y/N)	Incluir datos en la importación.
SHOW (Y/N)	Si <i>SHOW=Y</i> , el contenido del fichero de "export" se lista pero no se importa. Solo puede usarse con los parámetros <i>FULL=Y</i> , <i>FROMUSER</i> , <i>TOUSER</i> , o <i>TABLES</i> .
TABLES	Lista de tablas a importar.
TABLESPACES	Lista de espacios de almacenamiento a importar.
TO_USER	Lista de esquemas donde importar. Se requiere <i>IMP_FULL_DATABASE</i> .
TRANSPORT_TABL SPACE	Importar datos de espacios de almacenamiento.

“TABLESPACES” TRANSPORTABLES.

- Permite trasladar un conjunto de espacios de almacenamiento desde una bd a otra (sólo entre bd con el mismo juego de caracteres -character set- y sobre plataformas compatibles).
- Los tbsp transportados pueden ser gestionados localmente o no. No necesitan ser del mismo tamaño de bloque que el tamaño de bloque estándar de la bd destino.

Es util, por ejemplo, para:

- Realizar una recuperación a un punto en el tiempo “Tablespace Point-in-Time Recovery” (TSPITR).
- Mover índices (evitando así una reconstrucción necesaria si se importaran o cargaran datos).

“TBSP” TRANSPORTABLES. LIMITACIONES.

- Bd fuente y destino deben tener misma plataforma HW.
- Ambas bbdd deben usar el mismo juego de caracteres -character set- y el mismo juego de caracteres nacional -national character set-.
- No puede transportarse un tbsp a una bd destino en el cual exista un tbsp con el mismo nombre.
- Los tbsp transportables no soportan:
 - Vistas materializadas/replicación
 - Function-based indexes
 - Scoped REFs
 - Advanced Queuing

“TBSP” TRANSPORTABLES. COMPATIBILIDAD.

- En una bd con un tbsp SYSTEM gestionado localmente, los tbsp de diccionario no pueden crearse. Es posible trasladar un tbsp gestionado por diccionario pero no puede habilitarse para escritura.
- El parámetro compatible en ambas bbdd debe ser 8.1 o superior. Si el tamaño de bloque del tbsp transportado es diferente del de la bd destino, COMPATIBLE debe ser 9.0 o superior. No es necesario ejecutar la misma version de Oracle en las bbdd origen -mínimo versión 8i- y destino.

Al crear los tbsp transportables, Oracle calcula el nivel de compatibilidad más bajo al cual la bd destino debe estar -nivel de compatibilidad-. Al migrar, se produce error si no se cumple la condición.

“TBSP” TRANSPORTABLES. VARIABLES.

- *TABLESPACES*

Valor por defecto: none. Indica que todas las tablas en el tbsp se exportarán, incluyendo todas aquellas que tienen una partición localizada en él. Los índices son exportados con sus tablas, independientemente de donde se almacenan.

- *TRANSPORT_TABLESPACE*

Valor por defecto: n. Cuando se indica “y”, permite la exportación de metadatos correspondiente al tbsp.

"TBSP" TRANSPORTABLES. PROCESO 1/5.

- *Seleccionar un conjunto de tbsp "autocontenidos".*
- No deben haber dependencias entre objetos del conjunto de tbsp transportables y aquellos que no sean del mismo. Se incumple si, por ejemplo, existe:
 - Un índice dentro del conjunto está definido para una tabla fuera del mismo (sí puede existir un índice fuera del conjunto definido para una tabla del mismo).
 - Una tabla particionada parcialmente contenida en el conjunto.
 - Una "constraint" de integridad referencial apuntando a una tabla (al transportar tbsp pueden incluirse restricciones de integridad o no).
 - Una tabla con una columna LOB que referencia LOBS fuera del conjunto.

“TBSP” TRANSPORTABLES. DBMS_TTS.

- Para determinar si un conjunto de tbsp es autocontenido se emplea el procedimiento *TRANSPORT_SET_CHECK* del paquete *DBMS_TTS* (debe poseerse el rol *EXECUTE_CATALOG_ROLE*).

```
DBMS_TTS.TRANSPORT_SET_CHECK (  
    ts_list          IN VARCHAR2,  
    incl_constraints IN BOOLEAN DEFAULT,  
    full_closure    IN BOOLEAN DEFAULT FALSE);
```

ts_list ... Lista de tbsp, separados por coma.

incl_constraints ... *TRUE* para considerar las restricciones de integridad referencial en el análisis (valor por defecto).

full_closure ... Por defecto *FALSE*. Determinar o no si está autocontenido estrictamente (dependencias desde-hacia).

"TBSP" TRANSPORTABLES. DBMS_TTS.

- Para una aplicación estricta, el parámetro TTS_FULL_CHECK debe tener el valor TRUE (por ejemplo, Tablespace Point-in-Time Recovery -TSPITR-).
- Todas las excepciones producidas pueden verse a través de la vista TRANSPORT_SET_VIOLATIONS. Si la vista está vacía, el conjunto de tbsp es autocontenido.

```
SQLPLUS> EXECUTE DBMS_TTS.TRANSPORT_SET_CHECK  
('NOMINA,PAGOS', TRUE);
```

```
SQLPLUS> SELECT * FROM TRANSPORT_SET_VIOLATIONS;
```

"TBSP" TRANSPORTABLES. PROCESO 2/5.

- Generar un conjunto de tbsp. transportables.

- Cambiar el estado de los tbsp a sólo lectura.

```
ALTER TABLESPACE <nombre_tbsp> READ ONLY;
```

- Realizar la exportación de los tbsp a migrar (sólo se exportan metadatos) -conexión con privilegio SYSDBA-.

```
TRANSPORT_TABLESPACE=y  
TABLESPACES=(NOMINA, PAGOS)  
TRIGGERS=y  
CONSTRAINTS=n  
GRANTS=n
```

"TBSP" TRANSPORTABLES. PROCESO 2/5.

Si el autocontenido debe ser estricto:

```
TRANSPORT_TABLESPACE=y  
TABLESPACES=(NOMINA, PAGOS)  
TTS_FULL_CHECK=Y  
FILE=exportacion_tbsp.dmp
```

- TTS_FULL_CHECK es un parámetro de EXPORT que verifica que un conjunto de tbsp a recuperar no tiene dependencias (hacia/desde).
- Si el conjunto de tbsp no es autocontenido, la exportación falla. En dicho caso hay que resolver las dependencias.spacio al estado de lectura/escritura.

“TBSP” TRANSPORTABLES. PROCESO 3/5.

- Copiar los ficheros de datos correspondientes a los tbsp a transportar al lugar de destino. Hacer igual con el fichero resultado de la exportación.

"TBSP" TRANSPORTABLES. PROCESO 4/5.

- Integrar el conjunto de tbsp transportados.
 - Importar la información relativa a los tbsp. (utilidad *IMPORT*).

```
TRANSPORT_TABLESPACE=y  
FILE=exportacion_tbsp.dmp  
DATAFILES=('/u02/.../CURSOxx','/u02/.../CURSO')  
TABLESPACES=(NOMINA, PAGOS)  
TTS_OWNERS=(CONTABLE)  
FROMUSER=(CONTABLE)  
TOUSER=(ADMINISTRACION)
```

TTS_OWNERS hace referencia a los usuarios propietarios de los datos en el conjunto de tbsp migrado.

“TBSP” TRANSPORTABLES. PROCESO 5/5.

- Comprobar si se han generado errores en el proceso de importación.
- Cambiar el espacio al estado de lectura/escritura.

```
ALTER TABLESPACE <nombre_tbsp>READ WRITE;
```

“TBSP” TRANSPORTABLES. PROCESO 5/5.

- Si se migran tbsp de una tamaño de bloque distinto al tamaño de bloque estándar de la bd destino, en esta debe añadirse el parámetro de inicialización *DB_nK_CACHE_SIZE*.
- *DB_nK_CACHE_SIZE* (donde n = 2, 4, 8, 16, 32) es un parámetro dinámico (modificable con *ALTER SYSTEM*) que permite especificar el tamaño de la caché para nK buffers.
- Por ejemplo, si el tbsp a migrar tiene 8K de tamaño de bloque y la bd destino 4K, debe incluirse *DB_8K_CACHE_SIZE* como parámetro.

“TBSP” TRANSPORTABLES. RESUMEN PROCESO.

- Proceso.
 - Cambiar los espacios de almacenamiento al estado de sólo lectura.
 - Usar la utilidad export para trasladar la información almacenada en el diccionario (metadatos) a la bd destino.
 - Copiar los ficheros de datos correspondientes a dichos espacios de almacenamiento en el destino.
 - Usar la utilidad import para integrar la información en la bd destino.
 - Cambiar el espacio al estado de lectura/escritura.

VISTAS.

- *V\$DATAFILE.*
- *V\$TABLESPACES.*
- *V\$LOGFILE.*
- *V\$CONTROLFILE.*
- *V\$BACKUP.*
- *V\$INSTANCE.*
- *V\$ARCHIVED_LOG.*

TEMA 10.

RECUPERACION.

TEMA 10.

RECUPERACION.

- RECUPERACION.
- ESTRUCTURAS DE RECUPERACION.
- TIPOS DE RECUPERACION.
 - COMPLETA.
 - INCOMPLETA.
- ESTRATEGIAS DE RECUPERACION.
 - EN MODO NOARCHIVELOG.
 - EN MODO ARCHIVELOG.
- RESTAURACION DE FICHEROS.
- FICHEROS A RESTAURAR.
- RESTAURAR "REDO" ARCHIVADO.
- RESTAURAR FICHEROS DE COPIA.
- RESTAURAR Y RECREAR F. CONTROL.
 - PERDIDA DE MIEMBRO MULTIPLEXADO.
 - PERDIDA DE TODOS MIEMBROS.

TEMA 10.

RECUPERACION.

- PERDIDA DE MIEMBROS Y COPIAS.
- CONDICIONES RECUPERACION.
- RECUPERACION COMPLETA.
 - BD CERRADA.
 - BD ABIERTA.
- RECUPERACION INCOMPLETA.
 - “CANCEL BASED”.
 - “TIME BASED”.
 - “CHANGE BASED”.
- RECUPERACION PARALELA.
- APERTURA BD TRAS RECUPERACION.
- TSPITR (“Tbsp. point in time recovery”).
- VISTAS.

RECUPERACION.

- La posibilidad de un fallo de sistema o de maquina siempre existe. Si ocurre y afecta a la bd, esta debe recuperarse. El objetivo es asegurar que los efectos de las transacciones validadas se reflejen en la bd recuperada y la vuelta a una operación normal de la bd lo antes posible.
- Existen distintos tipos de fallos que pueden hacer necesario la recuperación:
 - Error de usuario. Por ejemplo, el borrado de una tabla (puede recuperarse de un *export* previo).
 - Fallo de una sentencia. Se deshace cualquier cambio automáticamente y se devuelve el control al usuario.

RECUPERACION.

- Fallo de proceso. P.ej. desconexión anormal. PMON lo detecta, hace "rollback" de transacción y libera recursos.
- Fallo de instancia. Debido a problemas de máquina, un fallo de s.o. ... Oracle realiza sin intervención del usuario la recuperación de la instancia ("crash recovery").
- Fallo en disco ("*media failure*"). "*Media recovery*". Deben restaurarse los ficheros dañados o perdidos desde copia usando sentencias de s.o. y realizar recuperación con la sentencia SQL*Plus *RECOVER* (modo ARCHIVELOG).

Los dos tipos de "media recovery" son *completa* -todo "redo" generado en la bd se aplica- o *incompleta* -no se aplica todo el "redo" generado y se obtiene una versión anterior de la bd- (en ambos casos afecta a la bd en su totalidad, al completo). *TSPITR*, "*tablespace point-in-time recovery*", permite realizar una recuperación incompleta de un tbsp concreto.

ESTRUCTURAS DE RECUPERACION.

- Redo Log.

Conjunto de ficheros que protegen los datos modificados en memoria que aun no han sido escritos a los ficheros de datos.

Consta del "redo log" en línea y el "redo log" archivado.

En modo ARCHIVELOG la bd puede recuperarse completamente de fallos de instancia y de disco. Tambien puede hacerse copia mientras está abierta y en uso.

En modo NOARCHIVELOG, la bd puede ser recuperada de un fallo de instancia pero no de disco. La bd sólo puede copiarse si está cerrada.

ESTRUCTURAS DE RECUPERACION.

- Registros de “undo”.

Almacenados en espacios de almacenamiento de “undo” o en “rollback segments”.

- Ficheros de control.

Guardan, entre otras cosas, información sobre la estructura de ficheros de la bd y el actual “log sequence number” escrito por LGWR. Esta información es usada durante la recuperación

- Copias de bd (“backups”).

Realizadas mediante Recovery Manager o con utilidades s.o..

TIPOS DE RECUPERACION. COMPLETA.

- Completa.

Se emplean datos de "redo" combinados con una copia de la bd, tbsp., o fichero de datos para actualizarlo al punto más actual posible. Se aplican todos los cambios anotados en el "redo".

Si se pretende recuperar la bd completa, debe:

- Montarse la bd.
- Asegurarse de que todos los ficheros a recuperar están en línea.
- Restaurar una copia de la bd o ficheros a recuperar.
- Aplicar "redo" (en línea, archivado o ambos).

TIPOS DE RECUPERACION. COMPLETA.

Si se pretende recuperar un tbsp o fichero de datos, debe:

- Si la bd está abierta, poner el tbsp o fichero fuera de línea.
- Restaurar una copia de los ficheros a recuperar.
- Aplicar "redo" (en línea, archivado o ambos).

TIPOS DE RECUPERACION. INCOMPLETA.

- Incompleta.

Se emplea una copia de la bd para generar una versión no actual de la bd. No se aplican todos los cambios posteriores a la última copia.

Se realiza sobre la bd al completo: *DBPITR* -db point-in-time recovery-. En los siguientes casos:

- Un fallo en disco destroza algunos o todos los "redo" en línea.
- Se ha producido una pérdida de datos, por ejemplo el borrado de una tabla (inconveniente tener "export"!).
- No puede realizarse una recuperación completa pues se han perdido "redo" archivados.
- Se han perdido los ficheros de control y debe usarse una copia para abrir la bd.

TIPOS DE RECUPERACION. INCOMPLETA.

- Para realizarla deben recuperarse todos los ficheros de datos precisos de copias anteriores a aquel momento al que se pretenda restaurar; y abrir la bd con la opción *RESETLOGS*.

Al indicar "*alter database open resetlogs;*" se reinicia el número de secuencia de "log" actual -número 1- y se desecha cualquier información de "redo" no aplicada durante la recuperación, pone a cero. Después de abrir la bd debe realizarse una copia completa.

- Un caso particular es la recuperación de un tbsp "point in time" (TSPTIR). Permite recuperar uno o más tbsp. a un momento en el tiempo distinto del resto de la bd.

ESTRATEGIAS DE RECUPERACION.

- Si la bd está en modo NOARCHIVELOG y hay un fallo en disco, "media failure", se debe restaurar la copia completa y consistente más reciente y abrir la bd (si se quisiera la bd actualizada hasta el momento del fallo, habría que introducir todos los cambios manualmente ¡imposible!).
- Si la bd está en modo ARCHIVELOG y los "redo" están disponibles, puede emplearse una recuperación completa o incompleta para reconstruir la bd.

ESTRATEGIAS DE RECUPERACION.

- Probar los mecanismos de copia y recuperación en un entorno apropiado.
- Planificar la respuesta a los distintos tipos de incidentes. Se atenderá al siguiente procedimiento general:
 - Determinar la información que se necesita recuperar y restaurar.
 - Restaurar copia de los ficheros y cualquier fichero de "redo" archivado que sea necesario (a nivel de s.o.).
 - Recuperar los ficheros.
 - Abrir la bd. Si se hace una recuperación incompleta o se ha restaurado una copia del fichero de control, debe abrirse con la opción *RESETLOGS*.

ESTRATEGIAS NOARCHIVELOG.

Ficheros dañados	Estrategia
Ficheros de datos. (CERRADA)	Restaurar la bd completa desde copia consistente (los cambios posteriores a esta se pierden). Abrir la bd con opción <i>RESETLOGS</i> .
Ficheros de datos y "redo" en línea. (CERRADA)	Restaurar la bd completa desde copia consistente (los cambios posteriores a esta se pierden). Abrir la bd con opción <i>RESETLOGS</i> .
Ficheros de datos y todos los ficheros de control. (CERRADA)	Restaurar la bd completa desde copia consistente (los cambios posteriores a esta se pierden). Abrir la bd con opción <i>RESETLOGS</i> .

ESTRATEGIAS ARCHIVELOG.

Ficheros dañados	Estrategia
Uno o más ficheros de datos. (ABIERTA)	Realizar la recuperación del tbsp o fichero con la bd abierta. Los tbsp o ficheros se ponen fuera de línea, recuperados y puestos en línea. La bd permanece accesible durante el proceso.
Todos los ficheros de datos. (CERRADA)	Restaurar los ficheros de datos, montar el fichero de control y recuperar la bd completa. Suponiendo que los "redo" están disponibles, puede abrirse la bd sin hacer <i>RESETLOGS</i> .
Uno o más ficheros de datos y "redo" archivado necesario. (ABIERTA)	Realizar TSPITR en los tbsp hasta el punto del último "redo" archivado disponible.

ESTRATEGIAS ARCHIVELOG.

Ficheros dañados	Estrategia
Todo f.control y uno o más f.datos. (NO ABIERTA)	Restaurar los f.control y f.datos desde copia y recuperar los f.datos. No se pierden datos, pero la bd no está disponible durante la recuperación. Abrir la bd con <i>RESETLOGS</i> .
Todo f.control y uno o más f.datos, así como "redo" en línea o archivado preciso para recuperación. (NO ABIERTA)	Restaurar los ficheros necesarios de la copia y realizar una recuperación incompleta de la bd al punto del "log" disponible más reciente. Se pierden los datos contenidos en los "log" perdidos y en todos los posteriores. Abrir la bd con <i>RESETLOGS</i> .

RESTAURACION DE FICHEROS.

- Restaurar un fichero es reemplazarlo por una copia previa del mismo. Es necesario guardar la localización de los ficheros de datos, control, "redo" en línea, "redo" archivado, inicialización, comunicaciones y contraseñas.
- Localización datos, control y "redo" en línea.

```
SELECT NAME FROM V$DATAFILE  
UNION ALL  
SELECT MEMBER FROM V$LOGFILE  
UNION ALL  
SELECT NAME FROM V$CONTROLFILE;
```

RESTAURAR FICHEROS DE COPIA.

- Localización "redo" archivado.

```
SELECT NAME, VALUE FROM V$PARAMETER  
WHERE NAME LIKE log_archive_dest%  
AND VALUE IS NOT NULL ;
```

- Formato de "redo" archivado.

```
SHOW PARAMETER LOG_ARCHIVE_FORMAT
```

- Listado de "redo" archivado.

```
SELECT NAME FROM V$ARCHIVED_LOG;
```

FICHEROS A RESTAURAR.

- La vista *V\$RECOVER_FILE* muestra el estado de los ficheros que necesitan recuperación, el motivo, el SCN donde debe comenzar, ...

```
SELECT * FROM V$RECOVER_FILE;
```

Nota:La vista no es util si el fichero de control en uso es una copia restaurada o uno nuevo creado tras el fallo en disco ("media failure").

- De *V\$DATAFILE* y *V\$TABLESPACE* pueden obtenerse nombres de fichero y de "tablespaces" de aquellos ficheros que necesitan recuperación.

```
SELECT r.FILE# AS df#, d.NAME AS df_name, t.NAME AS  
      tbsp_name, d.STATUS, r.ERROR, r.CHANGE#, r.TIME  
FROM V$RECOVER_FILE r, V$DATAFILE d, V$TABLESPACE t  
WHERE t.TS# = d.TS# AND d.FILE# = r.FILE#
```

RESTAURAR "REDO" ARCHIVADO.

- El "redo" archivado debe ser accesible durante la recuperación.
- Para averiguar que ficheros de "redo" son necesarios, se consultan las vistas:
 - V\$ARCHIVED_LOG ... Nombres de fichero de "redo" archivado.
 - V\$RECOVERY_LOG ... "redo" archivado necesario para la recuperación (sólo contiene información en caso de necesitarse recuperación para un fichero de datos, no si se trata de una recuperación planificada por un fallo de usuario).

RESTAURAR FICHEROS DE COPIA.

- Determinar los ficheros de datos a recuperar.
- Si la bd está abierta, poner los "tablespaces" fuera de línea:

```
ALTER TABLESPACE users OFFLINE IMMEDIATE;
```

- Restaurar copias de los ficheros dañados a su misma localización:

```
cp /copia/tbs_24.bak /u02/.../tbs_24.f
```

- Recuperar (*RECOVER*) y poner el tbsp. recuperado en línea.

RESTAURAR Y RECREAR F. CONTROL.

- Si un fallo de disco, "media failure", afecta a los ficheros de control (multiplexados o no), la bd sigue trabajando hasta la primera vez que un proceso "background" necesita acceder a los mismos; en este instante la bd se cierra automáticamente.
- Si el fallo es temporal y la bd todavía no se ha cerrado, puede intentarse corregir el fallo. Si la bd cierra, puede arrancarse tras corregir el problema y restaurar el acceso a los ficheros de control.

RECUPERAR F. CONTROL. PERDIDA DE MIEMBRO MULTIPLEXADO.

- En el caso de que un fallo haya dañado uno o más ficheros de control y, al menos, exista un f.control no dañado.
- Suponiendo que el disco y el sistema de ficheros donde reside el fichero dañado funcionan correctamente:
 - Parar la bd: *SHUTDOWN ABORT*
 - Corregir el problema hw.
 - Copiar un fichero de control intacto sobre el/los ficheros/s de control dañado/s

```
cp /u02/.../control01.ctl /u03/.../control02.ctl
```

- Arrancar la bd: *STARTUP*

RECUPERAR F. CONTROL. PERDIDA DE MIEMBRO MULTIPLEXADO.

- Si el disco y el sistema de ficheros donde reside el fichero dañado NO funcionan correctamente:
 - Parar la bd: *SHUTDOWN ABORT*
 - Copiar el fichero de control intacto a una localización alternativa: *cp /u02/.../control01.ctl /u05/.../control02.ctl*
 - Editar el fichero de parámetros de inicialización y modificar la variable *CONTROL_FILES* de forma que contenga SOLO las localizaciones actuales de los ficheros de control válidos.

```
CONTROL_FILES      =      '/u02/.../control01.ctl',  
                    '/u05/.../control02.ctl'
```

- Arrancar la bd: *STARTUP*

RECUPERAR F. CONTROL. PERDIDA DE TODOS MIEMBROS.

- No puede montarse ni abrirse la bd hasta tener un fichero de control accesible. Si se intenta montar la bd, se produce el error *"ORA-00205: error in identifying controlfile, check alert log for more info"*.
- Al restaurar una copia del fichero de control, debe abrirse la bd con la opción RESETLOGS.
- Si se tiene copia del fichero de control.
- Suponiendo que el disco y el sistema de ficheros donde reside el fichero dañado funcionan correctamente:
 - Parar la bd: *SHUTDOWN ABORT*
 - Corregir el problema hw.

RECUPERAR F. CONTROL. PERDIDA DE TODOS MIEMBROS.

- Restaurar la copia del fichero de control a todos los destinos indicados en el parámetro de inicialización *CONTROL_FILES* (orden cp).
- Montar la bd: *STARTUP MOUNT*
- Recuperar, sentencia *RECOVER* con la cláusula *USING BACKUP CONTROLFILE*. Indicar *UNTIL CANCEL* si se está realizando una recuperación incompleta:

RECOVER DATABASE USING BACKUP CONTROLFILE UNTIL CANCEL

- Aplicar los "log" archivados. Si se recibe un mensaje indicando que alguno no existe, probablemente significa que la información necesaria está en los "redo" en línea -cambios no archivados al suceder el problema-.

RECUPERAR F. CONTROL. PERDIDA DE TODOS MIEMBROS.

*ORA-00279: change 55636 generated at 06/08/2000 16:59:47 needed
for thread 1*

ORA-00289: suggestion : /oracle/work/arc_dest/arcr_1_111.arc

ORA-00280: change 55636 for thread 1 is in sequence #111

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

Puede indicarse el nombre de un "log" en línea:

/oracle/dbs/t1_log1.f

Log applied.

Media recovery complete.

Si los "redo" en línea no están accesibles, puede cancelarse la recuperación sin aplicarlos. En este caso debe recrearse el fichero de control.

- Abrir la bd con la opción *RESETLOGS: ALTER DATABASE OPEN RESETLOGS;*
- Realizar una copia completa de la bd.

RECUPERAR F. CONTROL. PERDIDA DE TODOS MIEMBROS.

- Si no puede restaurarse el fichero de control a su localización original; debe indicarse una nueva localización en el fichero de parámetros, variable *CONTROL_FILES*.
- La restauración es idéntica al caso en que se restaure a la localización por defecto; salvo que debe modificarse el fichero de parámetros de forma previa a la misma y al montaje de la bd.

RECUPERAR F. CONTROL. PERDIDA DE MIEMBROS Y COPIAS.

- Si los ficheros de control se han dañado pero el "redo" en línea permanece, puede recuperarse la bd tras crear un fichero de control. No se requiere abrir la bd, en este caso, con la opción RESETLOGS.

- Arrancar la bd , no montar: *STARTUP NOMOUNT*
- Crear el fichero de control, indicando la opción *NORESETLOGS*. Por ejemplo:

```
CREATE CONTROLFILE REUSE DATABASE SALES NORESETLOGS ARCHIVELOG  
...
```

Tras crear el fichero de control, Oracle monta la bd.

- Recuperar la bd: *RECOVER DATABASE*
- Abrir la bd: *ALTER DATABASE OPEN;*
- Realizar copia del fichero de control: *ALTER DATABASE BACKUP CONTROLFILE TO '/.../copia_fcontrol' REUSE;*

RECUPERAR F. CONTROL. PERDIDA DE MIEMBROS Y COPIAS.

- Existen distintas opciones para crear el fichero de control:
 - En el caso de haber ejecutado un *ALTER DATABASE BACKUP CONTROLFILE TO TRACE NORESETLOGS* tras el último cambio estructural, o antes de haber realizado este cambio; puede usarse el fichero de traza generado (en caso de no reflejar los últimos cambios, debe editarse y añadirlos).
 - Si se ha realizado una copia a fichero binario, *ALTER DATABASE BACKUP CONTROLFILE TO fichero;*, debe copiarse el fichero, montar la bd *-STARTUP MOUNT-* y realizar una copia a traza *-ALTER DATABASE BACKUP CONTROLFILE TO TRACE NORESETLOGS-*. Se edita, si es necesario, y se realizan los cambios oportunos.

RECUPERAR F. CONTROL. PERDIDA DE MIEMBROS Y COPIAS.

- Si no existe copia del fichero de control, debe crearse manualmente (sentencia *CREATE CONTROLFILE*).

CONDICIONES RECUPERACION.

- Para realizar cualquier tipo de "media recovery" debe tenerse en cuenta que:
 - Deben tenerse privilegios de administrador.
 - Una sesión no puede comenzar una recuperación completa mientras otra realiza una recuperación tipo incompleta.
 - No puede realizarse "media recovery" si se está conectado a la bd a través de un proceso servidor compartido ("shared server").
- Debe usarse la sentencia SQL*Plus *RECOVER*, y realizar una recuperación automática; así no se interroga al aplicar cada "log" archivado. Los nombres de estos, se obtienen concatenando *LOG_ARCHIVE_FORMAT* y *LOG_ARCHIVE_DEST_n*, siendo n el mayor valor entre los destinos habilitados.

CONDICIONES RECUPERACION.

- Hay dos formas de automatizar la aplicación de los “redo” necesarios en la recuperación:
 - Ejecutando *SET AUTORECOVERY ON* antes de *RECOVER*.

```
Startup mount  
Set autorecovery on  
Recover database  
...
```

- Indicando la cláusula *AUTOMATIC* como opción de la sentencia *RECOVER*.

```
Startup mount  
Recover automatic database  
...
```

RECUPERACION COMPLETA.

- Se recuperan las copias al SCN actual; bien la bd al completo, o tbsp o ficheros de datos individuales.
- No es necesario abrir la bd con RESETLOGS, de forma que pueden recuperarse ciertos ficheros de datos en un momento y los restantes posteriormente.

RECUPERACION COMPLETA. BD CERRADA.

1.- Preliminares.

- Cerrar la bd: SHUTDOWN ABORT.
- Intentar corregir el fallo que origina el error (puede que el fallo sea temporal y los datos no hayan sido dañados).

2.- Restaurar copias de los ficheros dañados o perdidos.

- Determinar los ficheros de datos a recuperar.
- Identificar la copia más reciente de los ficheros dañados (no restaurar ficheros de datos no afectados ni "redo" en línea).
- Restaurar los ficheros.

RECUPERACION COMPLETA. BD CERRADA.

3.- Recuperar la bd.

- Montar la bd: `STARTUP MOUNT.`
- Obtener los nombres de los ficheros de datos y su estado:
`SELECT NAME,STATUS FROM V$DATAFILE;`
- Asegurarse de que todos los f.datos de la bd están en línea: `ALTER DATABASE DATAFILE '/.../tbs.dbf' ONLINE;`
- Ejecutar la sentencia para recuperar la bd -todos los ficheros dañados- , un tbsp. concreto o un fichero de datos (puede indicarse recuperación automática):

RECOVER DATABASE;

RECOVER TABLESPACE users;

RECOVER DATAFILE '/.../tbs.dbf';

- Abrir la bd: `ALTER DATABASE OPEN;`

RECUPERACION COMPLETA. BD ABIERTA.

- Puede ocurrir un fallo con la bd abierta, dejando los ficheros no dañados en línea y en condiciones de uso. Oracle automaticamente pone los ficheros dañados fuera de línea -no sus "tablespaces"- si el DBWR no puede escribir en ellos.

Las consultas que no pueden leer en dichos ficheros generan errores, pero Oracle no los pone fuera de línea por esta razón:

```
ERROR at line 1:
```

```
ORA-01116: error in opening database file 11
```

```
ORA-01110: data file 11: '/oracle/dbs/tbs_32.f'
```

```
ORA-27041: unable to open file
```

```
SVR4 Error: 2: No such file or directory
```

```
Additional information: 3
```

RECUPERACION COMPLETA. BD ABIERTA.

- El proceso no es aplicable si se dañan los ficheros del tbsp. SYSTEM, en este caso se cierra la bd automáticamente.

1.- Preliminares.

- Poner los tbsp que contienen los ficheros dañados fuera de línea: *ALTER TABLESPACE users OFFLINE TEMPORARY;*
- Intentar corregir, si es posible, el fallo que origina el error.

2.- Restaurar copias de los ficheros dañados o perdidos.

- No deben restaurarse copias de ficheros no dañados, "redo" en línea o ficheros de control. Si se han restaurado a una localización alternativa, deben renombrarse los ficheros de datos: *ALTER DATABASE RENAME FILE '/.../tbs1.dbf' TO '/.../tbs1.dbf';*

RECUPERACION COMPLETA. BD ABIERTA.

3.- Recuperar los tbsp. fuera de línea.

- Conectar con privilegios de administrador (SYSTEM).
- Recuperar todos los tbsp. afectados (aconsejable automáticamente):

```
RECOVER TABLESPACE users, sales
```

- Poner los tbsp. en línea:

```
ALTER TABLESPACE users ONLINE;  
ALTER TABLESPACE sales ONLINE;
```


RECUPERACION INCOMPLETA.

- Al no recuperar la bd al momento más actual, debe indicarse cuando terminar la recuperación. Existen diversas opciones:
 - *Time-based* ... Recupera datos hasta un instante concreto.
 - *Cancel-based* ... Recupera hasta el empleo de la sentencia *CANCEL* (no disponible con Recovery Manager).
 - *Change-based* ... Recupera hasta un SCN específico.
 - *Log sequence* ... Recupera hasta un "log sequence number" específico (sólo con Recovery Manager).

RECUPERACION INCOMPLETA.

1.- Preliminares.

- Realizar una copia completa de la bd como medida de precaución ante errores durante la recuperación.
- Cerrar la bd, si está abierta: SHUTDOWN ABORT
- Intentar corregir el fallo que origina el error.

2.- Restaurar una copia completa de la bd.

- Si el fichero de control actual no refleja la estructura de la bd en el tiempo al que se quiere recuperar, debe restaurarse una copia del mismo que lo haga.
- Restaurar copias de todos los ficheros de datos de la bd (previas al momento al que se quiere recuperar).
- Abrir una sesión Sqlplus como administrador.

RECUPERACION INCOMPLETA.

- Montar la bd:

```
STARTUP MOUNT
```

- Si algún fichero fue restaurado a una localización alternativa, indicarlo:

```
ALTER DATABASE RENAME FILE '/.../df2.dbf' TO  
'/.../df2.dbf';
```

- Obtener el nombre y estado de todos ficheros de datos:

```
SELECT NAME,STATUS FROM V$DATAFILE;
```

- Asegurarse de que todos los ficheros de la bd están en línea (a menos que un tbsp se hubiera puesto fuera de línea normalmente o el fichero formara parte de un tbsp de sólo lectura):

```
ALTER DATABASE DATAFILE '/..../tbs.dbf' ONLINE;
```

RECUPERACION INCOMPLETA. "CANCEL BASED".

- La recuperación finaliza al indicar "CANCEL" en lugar de un nombre de fichero "redo" archivado, o cuando todos estos han sido aplicados. Permite controlar el proceso de recuperación.
- Deben seguirse los siguientes pasos:
 - Conectarse a Sqlplus con privilegios de administrador:
sqlplus '/ AS SYSDBA'
 - Montar la bd: *STARTUP MOUNT*
 - Comenzar la recuperación: *RECOVER DATABASE UNTIL CANCEL*
Si se está usando una copia del f.de control debe indicarse: *RECOVER DATABASE UNTIL CANCEL USING BACKUP CONTROLFILE*

-

RECUPERACION INCOMPLETA. "CANCEL BASED".

- Continuar aplicando "redo" hasta llegar al último y cancelar ejecutando: *CANCEL*

Oracle devuelve un mensaje indicando si la recuperación fue exitosa. Si se cancela una recuperación antes de que todos los ficheros esten en un SCN consistente y se intenta abrir la bd, se recibe un ORA-1113 en caso de ser necesaria más recuperación. Puede consultarse la vista *V\$RECOVER_FILE* para determinar si es así, o si una copia de algún fichero no fue restaurada antes de comenzar la recuperación.

- Abrir la bd en modo *RESETLOGS*. Siempre deben reiniciarse los "log" en línea después de una recuperación incompleta o con una copia del fichero de control:

ALTER DATABASE OPEN RESETLOGS;

RECUPERACION INCOMPLETA. "TIME BASED".

- Ejecutar `RECOVER DATABASE UNTIL TIME` para comenzar. El tiempo siempre se indica usando el formato: `'YYYY-MM-DD:HH24:MI:SS'`. Por ejemplo:

```
RECOVER DATABASE UNTIL TIME '2004-12-31:12:47:30'
```

Si se usa una copia del fichero de control, restaurada:

```
RECOVER DATABASE UNTIL TIME '2004-12-31:12:47:30'  
USING BACKUP CONTROLFILE
```

- Aplicar "redo" archivado. Si el fichero de control es una copia, tras aplicar "redo" archivado debe indicarse el "redo" en línea. Finaliza automáticamente la recuperación al alcanzar el tiempo indicado y se devuelve un mensaje indicando si es correcta.
- Abrir bd en modo "RESETLOGS": `ALTER DATABASE OPEN RESETLOGS;`

RECUPERACION INCOMPLETA. "CHANGE BASED".

- Indicar el SCN de terminación. Por ejemplo:

RECOVER DATABASE UNTIL CHANGE 10000;

- Aplicar "redo" archivado. Si el fichero de control es una copia, tras aplicar "redo" archivado debe indicarse el "redo" en línea. Finaliza automáticamente la recuperación al alcanzar el SCN indicado y se devuelve un mensaje indicando si es correcta.
- Abrir bd en modo "RESETLOGS": *ALTER DATABASE OPEN RESETLOGS;*

RECUPERACION PARALELA.

- La sentencia Sqlplus RECOVER PARALLEL indica recuperación paralela (por defecto es NOPARALLEL). Se selecciona un grado de paralelismo igual al número de UCP´s disponibles, relacionado con el parámetro de inicialización PARALLEL_THREADS_PER_CPU.

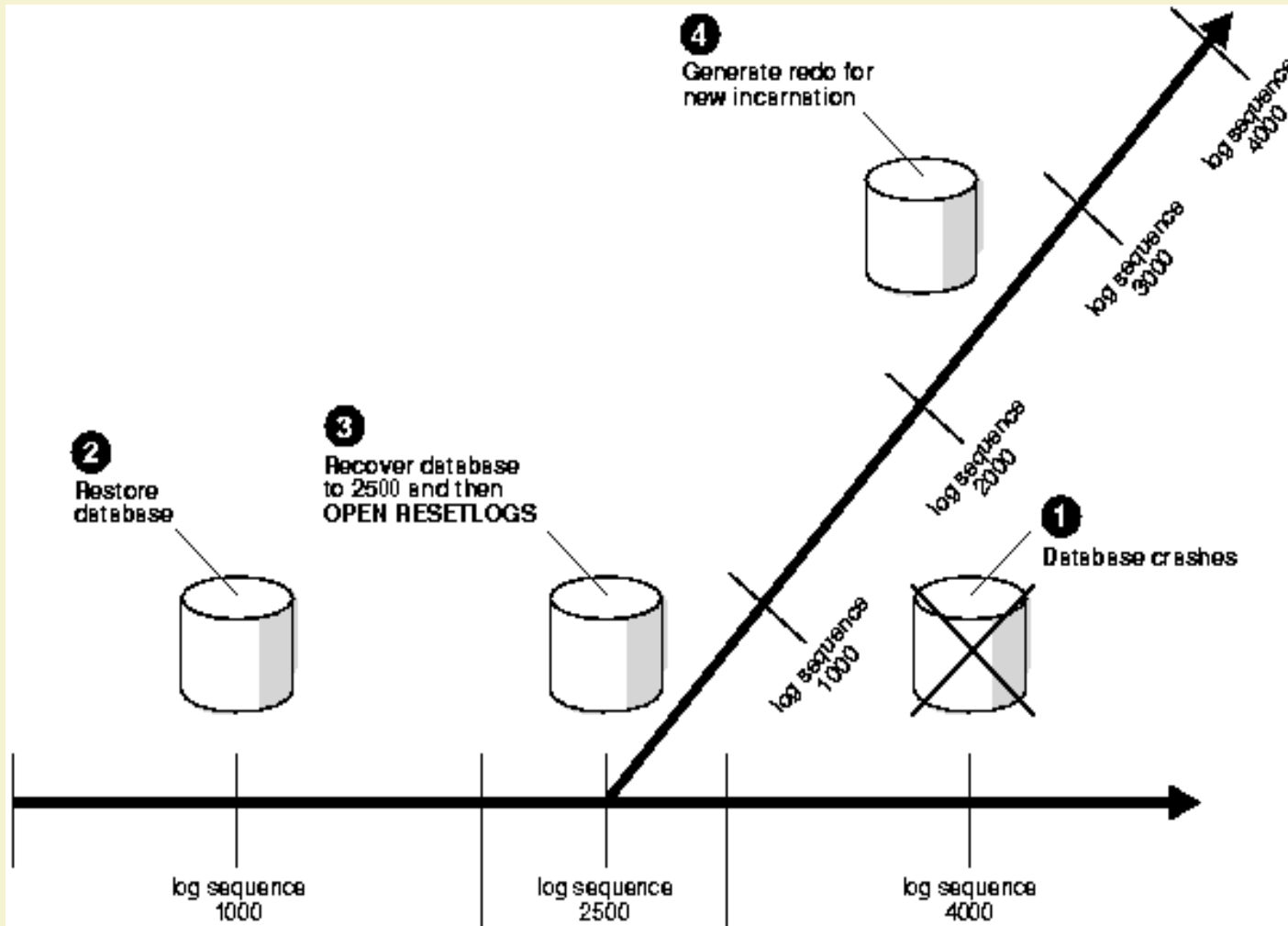
RECOVER PARALLEL <número>;

- Se fija así el número de procesos de recuperación usados para "media recovery". Si no se indica dicho número, Oracle selecciona un número por defecto de procesos de recuperación.

APERTURA BD TRAS RECUPERACION.

- Siempre que se realiza una recuperación incompleta o recuperación con una copia del fichero de control, debe reinicializarse los "logs" al abrir la bd. La nueva versión de bd se llama "incarnation". Todos los "redo" archivados generados tras el punto de RESETLOGS en la antigua "incarnation" son no válidos en la nueva.
- Al realizar una recuperación completa, no tiene que abrirse la bd con la opción RESETLOGS. Todos las copias y "redo" archivado creados durante la vida de esta "incarnation" de la bd son válidos.

APERTURA BD TRAS RECUPERACION.



APERTURA BD TRAS RECUPERACION.

- Al abrir la bd con RESETLOGS, todos los f.datos obtienen un nuevo RESETLOGS SCN y fecha, y la secuencia de "log" se reinicia a 1.
El "log" archivado contiene asimismo esos dos valores en la cabecera. Como Oracle no aplica un "log" archivado a un fichero de datos a menos que RESETLOGS SCN y fecha coincidan, esta operación evita que se corrompan con "log" archivados antiguos.
- Reinicializar los "logs" permite:
 - Descartar para siempre cualquier información no aplicada durante la recuperación.
 - Reinicializar la información en el fichero de control sobre "redo" en línea y "redo" threads.
 - Borrar el contenido del "redo" en línea y crear sus ficheros si no existían.
 - Reinicializar el número de secuencia de "log" a 1.

APERTURA BD TRAS RECUPERACION.

- Tras un RESETLOGS, es necesario:
 - Realizar una copia completa de la bd, cerrándola en modo normal (permite recuperar los cambios hechos después de la reinicialización).
 - Comprobar que en el alert_SID.log no existen entradas que reflejen inconsistencias detectadas entre el diccionario de datos y el fichero de control.

TSPITR (“Tbsp. Point in time recovery”).

- Junto con la funcionalidad de transporte de tbsp., permite recuperar uno o más tbsp. (distintos al SYSTEM) a un momento en el tiempo anterior al resto de la bd. Es util para:
 - Recuperar tras un borrado de tabla. Idem si está corrupta (ies conveniente tener “export”).
 - Recuperar tras un borrado accidental de un tbsp.
 - Recuperar tras sentencias que han afectado a una parte de la bd (DML).
 - Recuperar un esquema (si es independiente y reside en un tbsp aparte).
 - Recuperar un tbsp en una gran bd (VLDB -very large database-).

TSPITR.

- El desarrollo básico de un TSPITR consta de varios pasos:
 - Operaciones preliminares y puesta fuera de línea de aquellos tbsp. que requieren recuperación.
 - Creación de una bd auxiliar (duplicado de la bd en uso restaurada desde copia) y su recuperación hasta el momento necesario en el tiempo.

La bd auxiliar puede residir en la misma máquina que la bd en uso o en otra distinta. Se aconseja esta última opción por su mayor facilidad de manejo y menor probabilidad de daño.
 - Borrado de los tbsp que requieran TSPITR (bd en uso).
 - Transporte del conjunto de tbsp. desde la bd auxiliar hasta la bd en uso.

TSPITR. PRELIMINARES.

1.- Debe comprobarse que se poseen las copias necesarias previas al instante al que se desea el TSPITR; así como una copia del fichero de control realizada con ALTER DATABASE BACKUP CONTROLFILE TO '<fichero>';

Igualmente que hay suficiente espacio en disco y memoria para poder recuperar y arrancar la bd auxiliar.

2.- Obtener información de la bd en uso.

- Ficheros de datos, control y "redo" en línea.

```
SELECT NAME FROM V$DATAFILE UNION ALL  
SELECT MEMBER FROM V$LOGFILE UNION ALL  
SELECT NAME FROM V$CONTROLFILE;
```

TSPITR. PRELIMINARES.

- Ficheros de datos del tbsp SYSTEM y del conjunto de recuperación (tbsp a recuperar).

```
SELECT t.NAME AS "reco_tbs", d.NAME AS "dbf_name"  
      FROM V$DATAFILE d, V$TABLESPACE t  
      WHERE t.TS# = d.TS#  
      AND t.NAME IN ('SYSTEM', 'RECO_TBS_1', 'RECO_TBS_2');
```

- Ficheros de datos del tbsp de UNDO.

```
SELECT r.TABLESPACE_NAME AS "rbs_tbs", d.FILE_NAME AS  
      "dbf_name" FROM DBA_ROLLBACK_SEGS r, DBA_DATA_FILES d  
      WHERE r.TABLESPACE_NAME=d.TABLESPACE_NAME;
```

```
SELECT u.TABLESPACE_NAME AS "undo_tbs", d.FILE_NAME AS  
      "dbf_name" FROM DBA_UNDO_EXTENTS u, DBA_DATA_FILES d  
      WHERE u.TABLESPACE_NAME=d.TABLESPACE_NAME;
```


TSPITR. PRELIMINARES.

3.- Determinar los objetos que se perderán en el TSPITR.

Cualquier objeto creado tras el instante al que se desea recuperar se pierde. Para averiguar cuáles son , debe consultarse la vista *TS_PITR_OBJECTS_TO_BE_DROPPED* en la bd en uso.

Por ejemplo, si se quiere recuperar "ventas1" y "ventas2" al instante '2000-06-02:07:03:11':

```
SELECT OWNER, NAME, TABLESPACE_NAME, TO_CHAR
(CREATION_TIME, 'YYYY-MM-DD:HH24:MI:SS')
FROM SYS.TS_PITR_OBJECTS_TO_BE_DROPPED
WHERE TABLESPACE_NAME IN ('VENTAS1','VENTAS2')
AND CREATION_TIME > TO_DATE('00-JUN-02:07:03:11','YY-MON-
DD:HH24:MI:SS')
ORDER BY TABLESPACE_NAME, CREATION_TIME;
```

TSPITR. PRELIMINARES.

- 4.- Configurar la conexión a la bd auxiliar (Net9).
- 5.- Crear un fichero de parámetros de inicialización para la bd auxiliar.

Contendrá parámetros como: *DB_NAME*, *CONTROL_FILES*, *LOG_ARCHIVE_DEST_n*, *LOG_ARCHIVE_FORMAT*. Parámetros de memoria como *DB_CACHE_SIZE* , *SHARED_POOL_SIZE* y *LARGE_POOL_SIZE* deben mantenerse a un valor bajo.

TSPITR. CREACION BD AUXILIAR.

- Se asume que se creará en una máquina distinta a aquella en que reside la bd en uso, y se usarán los mismos nombres de ruta ("path names") para los ficheros.
- 1.- Restaurar los ficheros de la bd auxiliar (ficheros de control, undo y system, junto con el conjunto de recuperación -tbsp a recuperar-).
 - 2.- Arrancar la bd -sin montar-. Si es necesario indicar el fichero de parámetros de inicialización.

STARTUP NOMOUNT PFILE =

TSPITR. CREACION BD AUXILIAR.

- 3.- Montar la bd usando la cláusula CLONE (pone todos los ficheros de datos "offline" de forma automática).

```
ALTER DATABASE MOUNT CLONE DATABASE;
```

- 4.- Ejecutar la consulta siguiente para descartar cualquier error en el nombre de los ficheros de datos:

```
SELECT NAME FROM V$DATAFILE UNION ALL  
SELECT MEMBER FROM V$LOGFILE UNION ALL  
SELECT NAME FROM V$CONTROLFILE;
```

TSPITR. CREACION BD AUXILIAR.

5.- Poner todos los ficheros de datos en línea.

```
ALTER DATABASE DATAFILE /u02/.../system.dbf ONLINE;  
ALTER DATABASE DATAFILE /u02/.../ventas1.dbf ONLINE;  
ALTER DATABASE DATAFILE /u02/.../ventas2.dbf ONLINE;  
ALTER DATABASE DATAFILE /.../undo.dbf ONLINE;
```

6.- Recuperar la bd a un instante concreto con la opción *USING BACKUP CONTROLFILE*. Por ejemplo:

```
RECOVER DATABASE UNTIL CANCEL USING BACKUP  
CONTROLFILE
```

7.- Abrir la bd auxiliar con la opción *RESETLOGS*.

```
ALTER DATABASE OPEN RESETLOGS;
```

TSPITR. TRANSPORTAR TBSP. OPERACIONES BD AUXILIAR.

1.- Colocar los tbsp en modo sólo lectura.

```
ALTER TABLESPACE ventas1 READ ONLY;  
ALTER TABLESPACE ventas2 READ ONLY;
```

2.- Comprobar que el conjunto es autocontenido.

```
EXECUTE SYS.DBMS_TTS.TRANSPORT_SET_CHECK  
( 'ventas1,ventas2',TRUE,TRUE);  
  
SELECT * FROM SYS.TRANSPORT_SET_VIOLATIONS;
```

3.- Generar el conjunto transportable (utilidad Export).

TSPITR. TRANSPORTAR TBSP. OPERACIONES BD EN USO.

1.- Borrar los tbsp. del conjunto de recuperación.

```
DROP TABLESPACE ventas1 INCLUDING CONTENTS;  
DROP TABLESPACE ventas2 INCLUDING CONTENTS;
```

2.- Copiar los ficheros de datos de la bd auxiliar a la bd en uso, igual con el fichero resultado de la exportación (orden cp).

3.- Importar el conjunto transportado (utilidad import).

4.- Cambiar modo de los tbsp. transportados a lectura/escritura.

```
ALTER TABLESPACE ventas1 READ WRITE;  
ALTER TABLESPACE ventas2 READ WRITE;
```

TSPITR. TRANSPORTAR TBSP. OPERACIONES BD EN USO.

5.- Realizar una copia de los tbsp. recuperados.

Debe realizarse copia pues de otra forma podrían perderse. Si, por ejemplo, ocurre un fallo de disco, y se intenta recuperar desde una copia anterior al TSPITR, esta falla.

VISTAS.

- *V\$RECOVER_FILE* ... Muestra el estado de los ficheros que precisan "media recovery".