

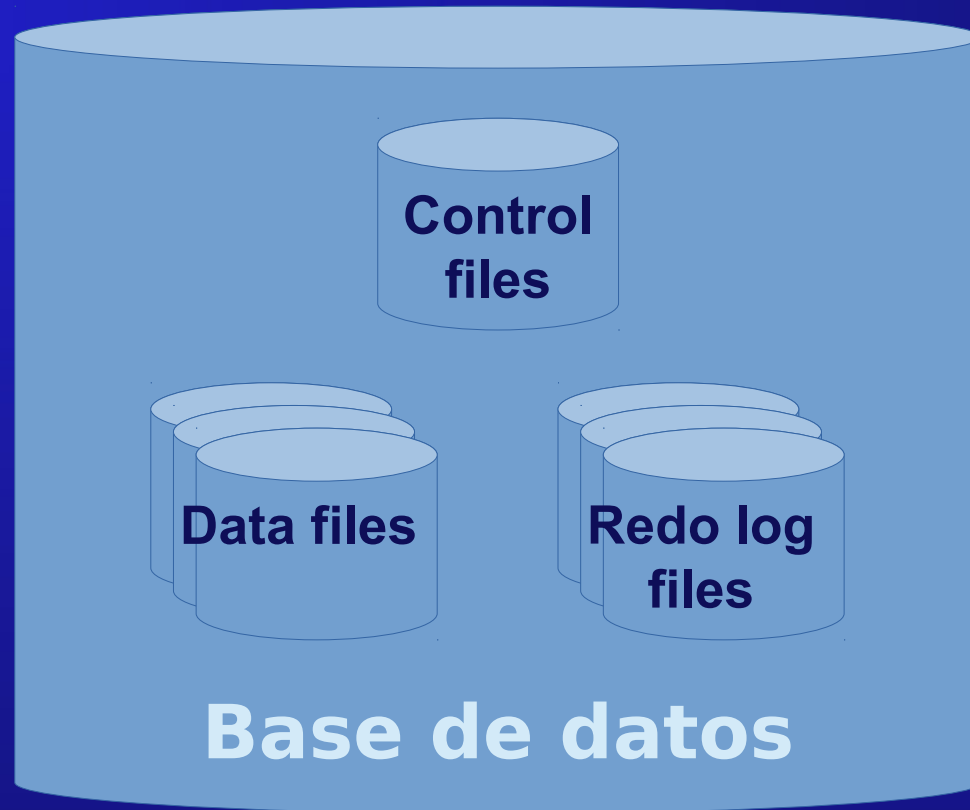
4 Estructura del Almacenamiento de Oracle

El fichero de Control (Control File)

Objetivos

- **Importancia y uso del archivo de control.**
- **Contenido del archivo de control.**
- **Obtención de información del archivo de control.**
- **Multiplexado del archivo de control.**

Control File



Archivos que se visualizan desde el S.O.

Contenido del Fichero de Control

- Nombre de la Base de Datos
- Localización de los Data files
- Localización de los Redo log files
- Nombres de los Tablespaces
- Número actual de registro de log
- Log History
- Information de backup

Parámetros que Afectan al Tamaño del Archivo de Control

- MAXLOGFILES: Num. max. de grupos de log
- MAXLOGMEMBERS: Num. max. de miembros por grupo
- MAXLOGHISTORY: Num. max. de archivos de redo log
- MAXDATAFILES: Num. max. de datafiles
- MAXINSTANCES: Num. max. de instancias que pueden abrir y montar la BD simultáneamente

**Todos ellos se especifican en el comando
CREATE DATABASE**

Información sobre el Fichero de Control

- **V\$CONTROLFILE**
 - NAME
- **V\$PARAMETER**
 - NAME (control_file)
 - VALUE

```
SELECT value FROM V$parameter  
WHERE name='control_files';
```

V\$<vista>: Vistas de rendimiento dinámico de la BD

Multiplexado del Fichero de Control

**control_files=(/DISK1/control01.ctl,/DISK2/control02.ctl)
(spfile<id>.ora)**

Disk 1



Disk 2



Multiplexado del Fichero de Control

1. Detener la instancia.
2. Copiar el control file en la nueva ubicación.
3. Cambiar el parámetro control_files en el pfile.
4. Levantar instancia de BD con ese pfile.
5. Crear SPFile a partir del pfile de inicio.
6. Detener la instancia de la BD.
7. Iniciar instancia con SPFile.

Ejercicios

1. Encontrar la localización del fichero de control y su nombre, usando `v$controlfile`, `v$parameter` y la Consola de Administración.
2. ¿Qué sucede si se arranca la BD sin ficheros de control ?
3. Crear el directorio
`/databases/app/ejercicios/control_files`
4. Poner en el directorio
`/databases/app/ejercicios/control_files`
una copia del fichero de control, asegurando que el servidor la tenga actualizada constantemente.

Ficheros de Recuperación (Redo Log Files)

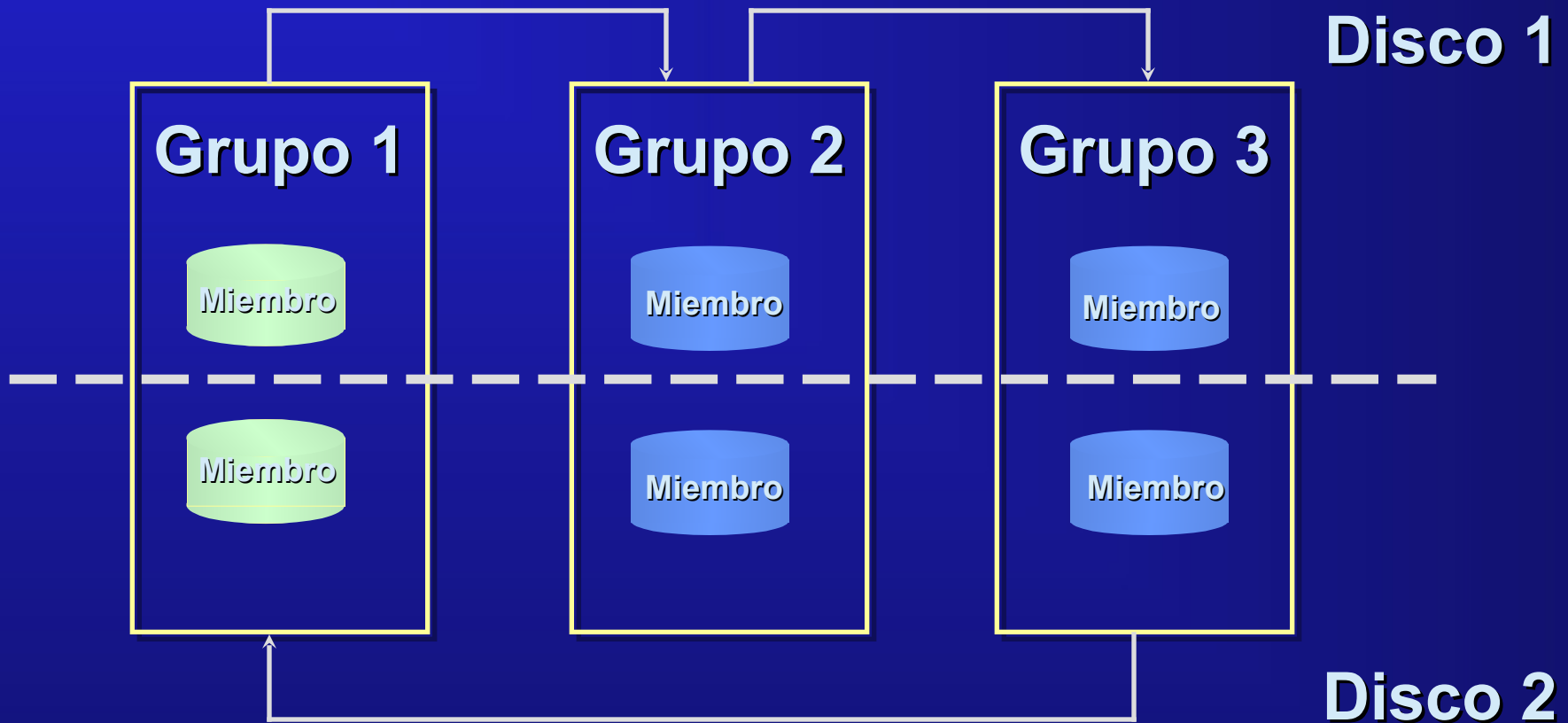
Objetivos

- **Uso de redo log files on-line.**
- **Información de log on-line y archivada.**
- **Control de cambio de log (switch) y checkpoints.**
- **Multiplexado y mantenimiento de on-line redo log files.**
- **Planificación de redo log files on-line.**

Utilidad de Redo Log Files

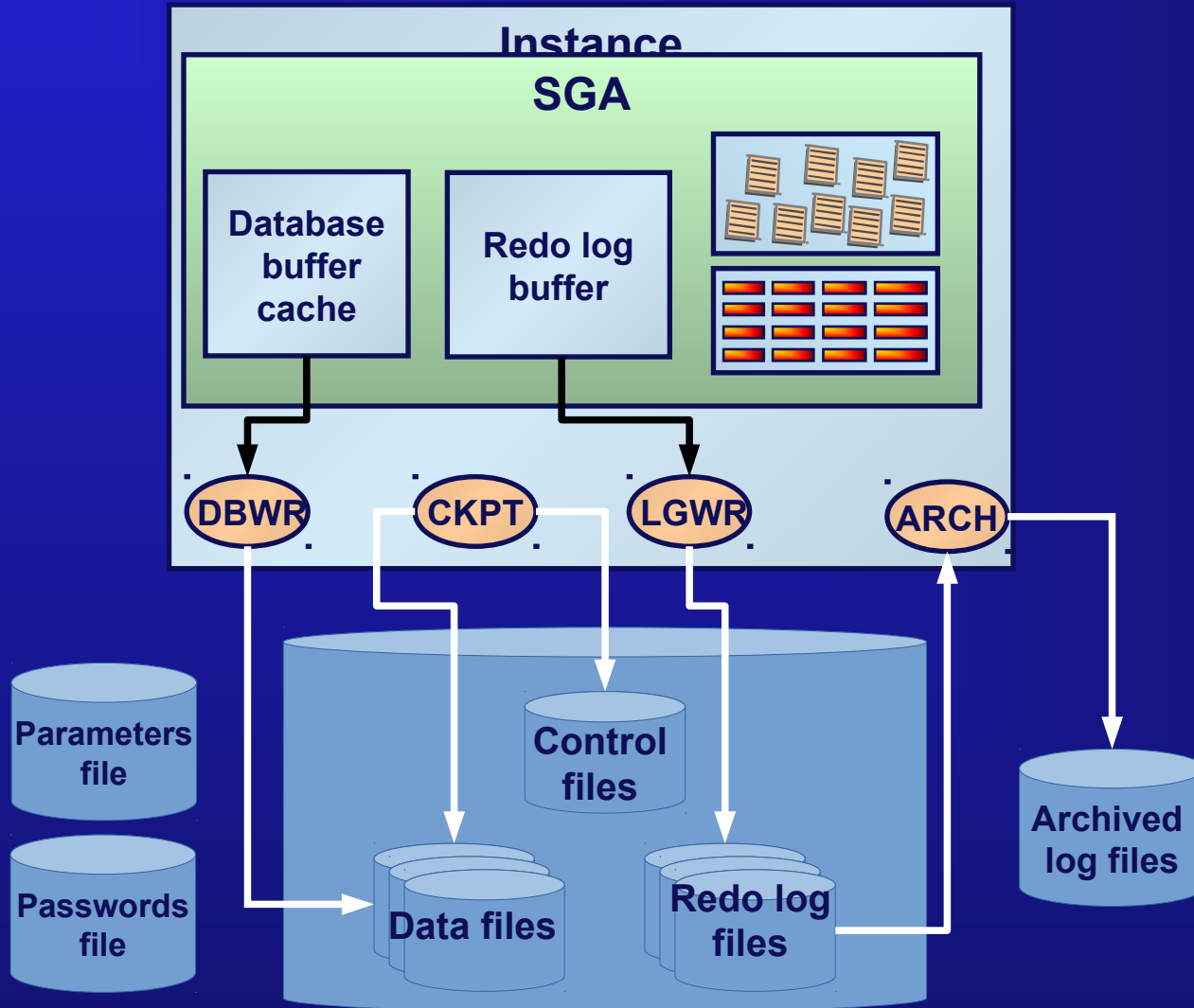
- 1) Anotar los cambios que se van produciendo en los datos.
- 2) Recuperar cambios que no han sido salvados en disco.
- 3) Se graban en disco (.log o .rdo) cuando:
 - Se ejecuta un *commit*.
 - Se llena un tercio del redo log buffer.
 - Cada tres segundos.
 - Antes de que DBWR escriba en disco un buffer de datos.

Grupos y Miembros de Redo Log



Hacen falta, como mínimo, dos grupos de redo log

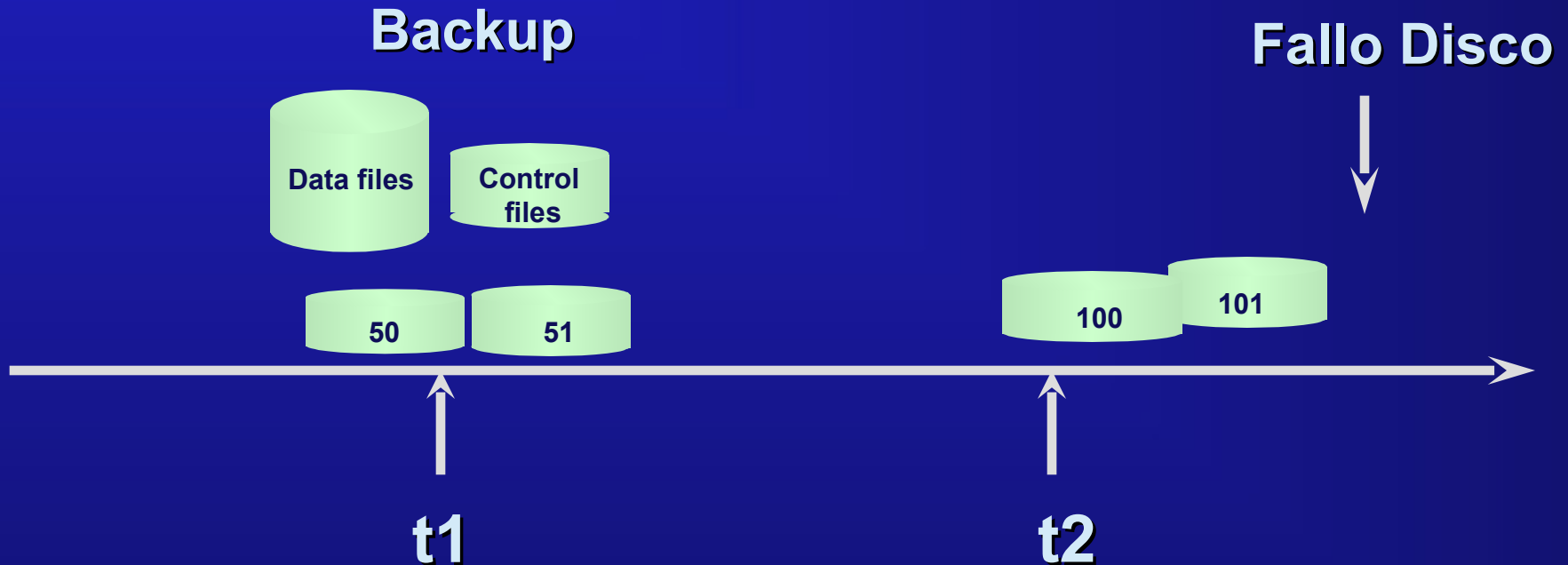
Arquitectura Oracle



Checkpoint

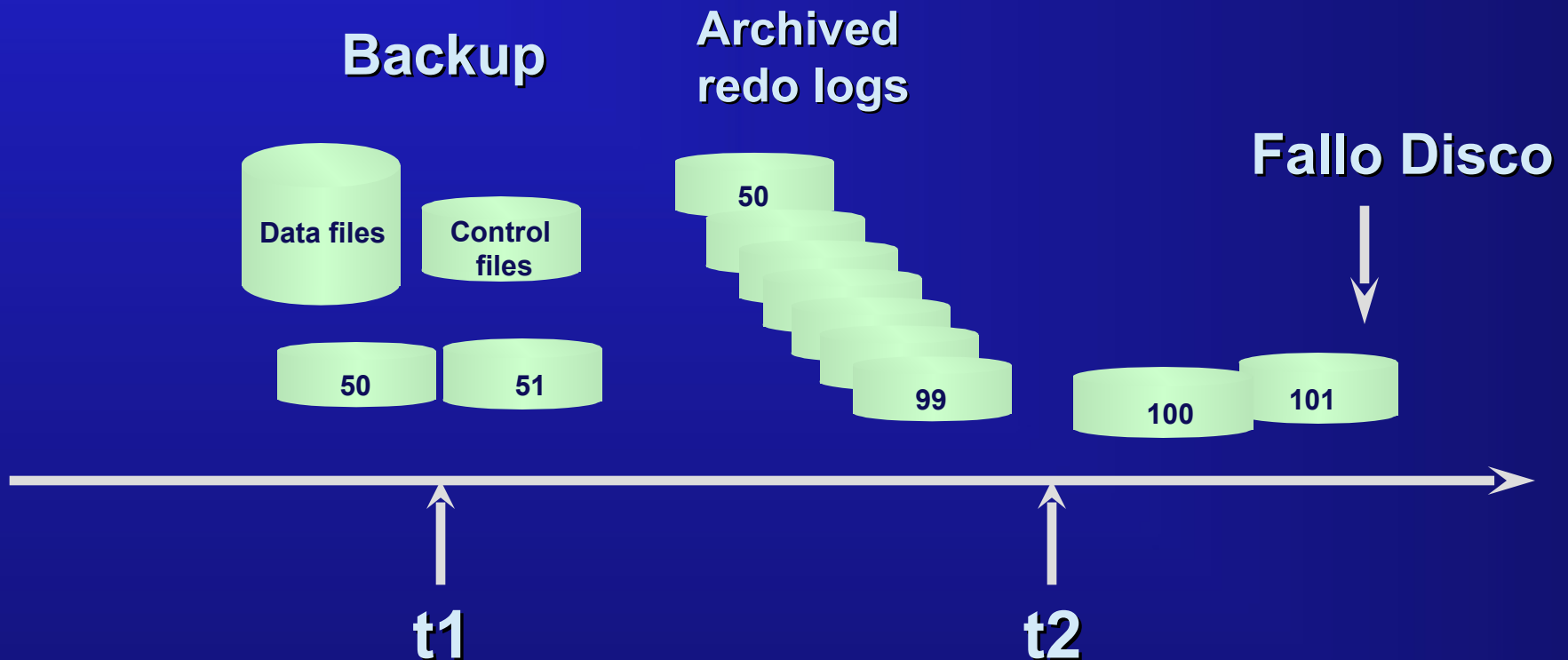
- *Log switch* es un cambio de grupo redo log.
- El administrador puede forzarlo.
- Cada *log switch* genera un *checkpoint*:
 - Se hacen efectivos los cambios en los datos.
 - El proceso CKPT anota este hecho en data y control files.
- Cuando el administrador “detiene” la BD en cualquiera de sus modos, también se lleva a cabo.

Redo Log sin Archivado



```
SQLPlus> ARCHIVE LOG stop;
```

Redo Log con Archivado



```
SQLPlus> ARCHIVE LOG start;
```

Información sobre Redo Log

```
ARCHIVE LOG LIST;
```

- **V\$DATABASE:**

- NAME
- LOG_MODE

- **V\$INSTANCE**

- ARCHIVER (stopped/started)

Database log mode

Automatic Archival

Archive destination

Oldest online log seq

Current online log seq

Información sobre Grupos y Miembros

V\$THREAD:

- **GROUPS**
- **CURRENT_GROUP#**
- **SEQUENCE#**

V\$LOG(buffers):

- **GROUP#**
- **MEMBERS**
- **STATUS**
- **SEQUENCE#**
- **BYTES**

V\$LOGFILE: archivos físicos

Información sobre Grupos y Miembros

```
SQLPlus> SELECT groups,current_group#,sequence#  
          FROM V$thread;
```

<u>GROUPS</u>	<u>CURRENT_GR</u>	<u>SEQUENCE</u>
2	1	689

```
SQLPlus> SELECT group#,sequence#,members,status  
          FROM V$log;
```

<u>GROUP#</u>	<u>SEQUENCE</u>	<u>MEMBERS</u>	<u>STATUS</u>
1	689	1	CURRENT
2	688	1	INACTIVE

Log Switches y Checkpoints

- Se puede forzar un cambio de log:

```
ALTER SYSTEM SWITCH LOGFILE;
```

- Parámetros de control de checkpoints:
 - **LOG_CHECKPOINT_INTERVAL:**
Fuerza checkpoint tras un nº de I/O
 - **LOG_CHECKPOINT_TIMEOUT:** Fuerza checkpoint cada n segundos

Añadir Grupos de Redo Log

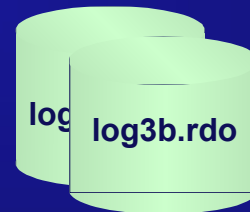
```
ALTER DATABASE ADD LOGFILE  
( '/oradata/.../log3a.rdo' ,  
  '/DISK4/log3b.rdo' ) size 1M;
```



Grupo 1



Grupo 2



Grupo 3

Añadir Miembros a Grupos

```
ALTER DATABASE ADD LOGFILE MEMBER  
  '/oradata/.../log1b.log' TO GROUP 1,  
  '/DISK4/log2b.log' TO GROUP 2;
```



Grupo 1



Grupo 2

Eliminación de Grupos

```
ALTER DATABASE DROP LOGFILE  
GROUP 3;
```



Grupo 1



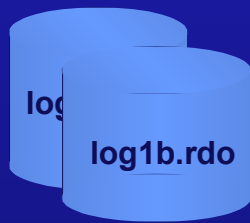
Grupo 2



Grupo 3

Eliminación de Miembros

```
ALTER DATABASE DROP LOGFILE MEMBER  
'/DISK4/log2b.log' ;
```



Grupo 1



Grupo 2

Ejercicios

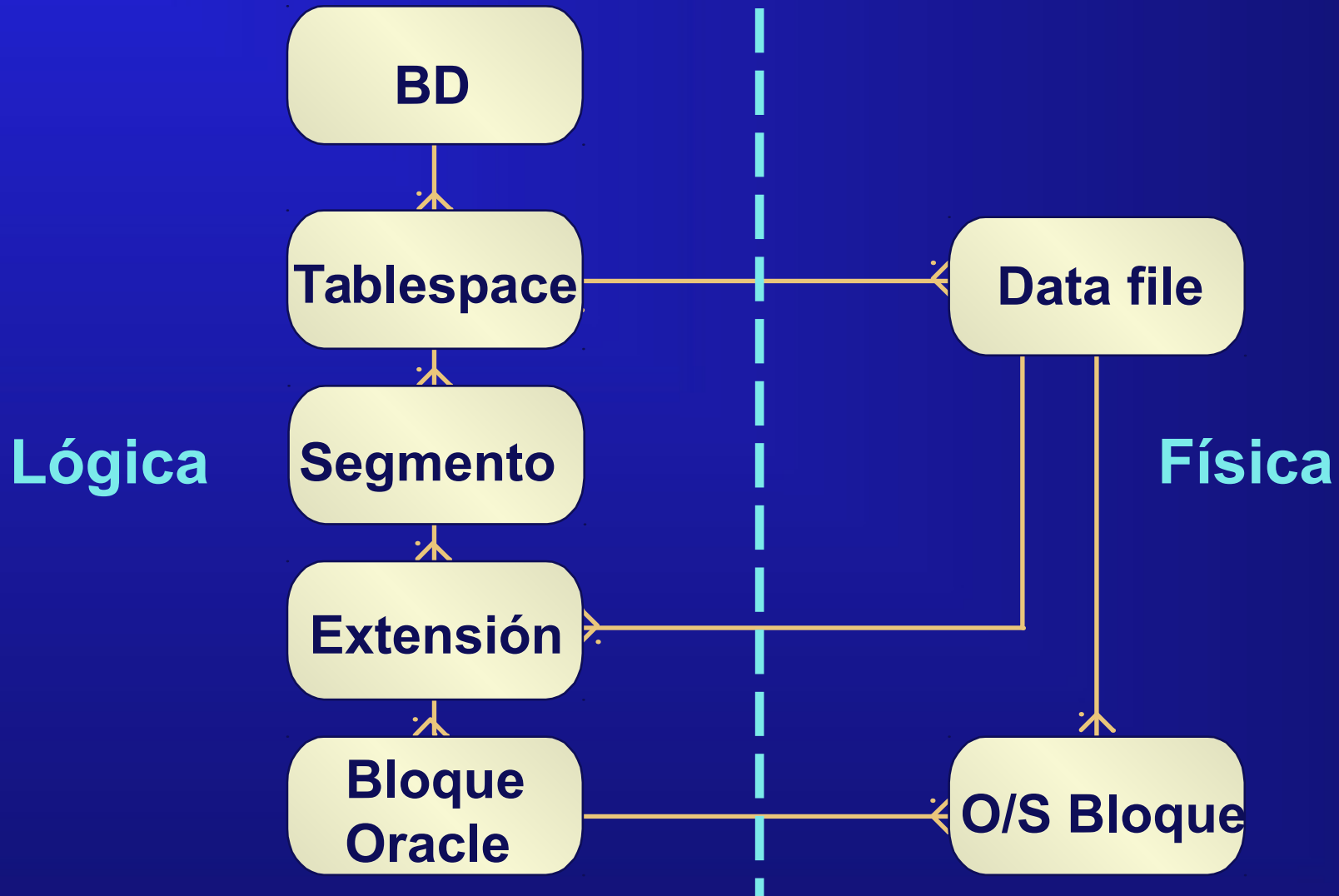
- 1. Listar número y localización de los log files y los grupos y miembros que hay.**
- 2. Encontrar el modo actual de redo.**
- 3. Añadir un miembro más a los grupos de redo log y verificar resultado.**
- 4. Crear un nuevo grupo de redo log.**

Tablespaces y Data Files

Objetivos

- **Descripción de la estructura lógica de la BD.**
- **Creación de tablespaces.**
- **Métodos para cambiar el tamaño de un tablespace.**
- **Cambiar el estado y el modo de almacenamiento de tablespaces.**
- **Localización de tablespaces.**
- **Necesidades y arquitectura ideal.**

Estructura de la BD



Tablespaces: Características

- **Un tablespace pertenece a una sólo BD.**
- **Contiene uno o más ficheros (datafiles).**
- **Se pueden habilitar y deshabilitar on-line (excepto SYSTEM).**
- **Pueden cambiar a modo sólo lectura on-line.**
- **Puede hacerse backup de un tablespace.**

Tablespaces SYSTEM y “USERS”

**Tablespace SYSTEM
contiene:**

- Información de catálogo
- Segmentos de rollback del SYSTEMA

**Tablespace “USERS”
contiene:**

- Segmentos Temporales
- Datos de usuarios
- Índices de usuarios

Creación de Tablespaces

```
CREATE TABLESPACE <tb_id>
DATAFILE '..\oradata/...\fich1.dbf' SIZE int K|M,
'/DISK5/<fich2>.dbf' SIZE int K|M]
[MINIMUM EXTENT n K|M]
[DEFAULT STORAGE (INITIAL int|NEXT int|
                    MAXEXTENTS n|MINEXTENTS n) ]
[PERMANENT | TEMPORARY]
[ONLINE | OFFLINE] ;
```

Creación de Tablespaces

Ejemplo:

```
CREATE TABLESPACE app_data
DATAFILE '/DISK4/app01.dbf' SIZE 10M,
        '/DISK5/app02.dbf' SIZE 10M
MINIMUM EXTENT 500K
DEFAULT STORAGE (INITIAL 500K NEXT 500K
                MAXEXTENTS 500) ;
```

Añadir Data Files a un Tablespace

```
ALTER TABLESPACE app_data  
ADD DATAFILE  
  '/DISK5/app03.dbf' SIZE 20M  
    AUTOEXTEND ON NEXT 10M  
      MAXSIZE 80M;
```

Tablespace APP_DATA



Alterar el Tamaño de un Data File

```
ALTER DATABASE DATAFILE
```

```
`/DISK5/app02.dbf' RESIZE 2M;
```

Tablespace APP_DATA



Cambiar Parámetros de Tablespace

```
ALTER TABLESPACE app_data  
MINIMUM EXTENT 2M;
```

```
ALTER TABLESPACE app_data  
DEFAULT STORAGE (INITIAL 2M  
NEXT 2M  
MAXEXTENTS 999) ;
```

Tablespace OFF-LINE

- No es accesible para los datos de los usuarios.
- El tablespace SYSTEM no puede inhabilitarse.
- El servidor ejecuta un checkpoint de todos los datafiles de ese tablespace.

```
ALTER TABLESPACE app_data OFFLINE;
```

Cómo mover Data Files con **ALTER TABLESPACE**

- El tablespace no es SYSTEM.
- El tablespace que contiene el fichero debe estar off-line.
- Los data files deben existir.

Ejemplo:

```
ALTER TABLESPACE app_data RENAME  
DATAFILE '/DISK4/app01.dbf' TO  
'/DISK5/app01.dbf' ;
```

Cómo mover Data Files con ALTER TABLESPACE

- 1. Poner el tablespace OFF-LINE.**
- 2. Mover o copiar los ficheros desde S.O.**
- 3. Ejecutar ALTER TABLESPACE RENAME DATAFILE.**
- 4. Poner el tablespace ON-LINE.**
- 5. Borrar (si se copiaron) los ficheros desde S.O.**

Cómo mover Data Files con ALTER DATABASE

- La BD debe estar montada (no abierta).
- El fichero debe existir.

Ejemplo:

```
ALTER DATABASE RENAME FILE  
  '/DISK1/system01.dbf' TO  
  '/DISK2/system01.dbf' ;
```

Cómo mover Data Files con ALTER DATABASE

- 1. Tirar la base de datos (shutdown).**
- 2. Copiar/mover los ficheros desde S.O.**
- 3. Montar la base de datos.**
- 4. Ejecutar ALTER DATABASE RENAME FILE.**
- 5. Abrir la BD (STARTUP OPEN).**

Tablespace modo Sólo-Lectura

- El tablespace debe estar on-line.
- No se permiten transacciones contra él.
- El tablespace no debe estar involucrado en un proceso de backup activo.

```
ALTER TABLESPACE app_data READ  
ONLY;
```

Eliminación de Tablespaces

- Un tablespace con datos no puede eliminarse sin la cláusula INCLUDING CONTENTS.
- Sólo se elimina su nombre del fichero de control (deben borrarse físicamente desde S.O.).
- Se recomienda ponerlo off-line para evitar que haya transacciones accediendo.

```
DROP TABLESPACE <tb_id>  
[INCLUDING CONTENTS [CASCADE CONSTRAINTS]] ;
```

Información sobre Tablespaces

DBA_TABLESPACES

- TABLESPACE_NAME
- NEXT_EXTENT
- MIN/MAX_EXTENTS
- MIN_EXTLEN
- STATUS (on-line/off-line)
- CONTENTS (temporary/permanent)

Información sobre Data Files

DBA_DATA_FILES

- **FILE_NAME**
- **TABLESPACE_NAME**
- **BYTES**
- **AUTOEXTENSIBLE**
- **MAXBYTES (limitación total)**

Información de Data Files y Tablespaces

V\$DATAFILE

- TS#
- NAME
- FILE#
- RFILE#
- STATUS
- ENABLED
- BYTES
- CREATE BYTES



V\$TABLESPACE

- TS#
- NAME

Ejercicios

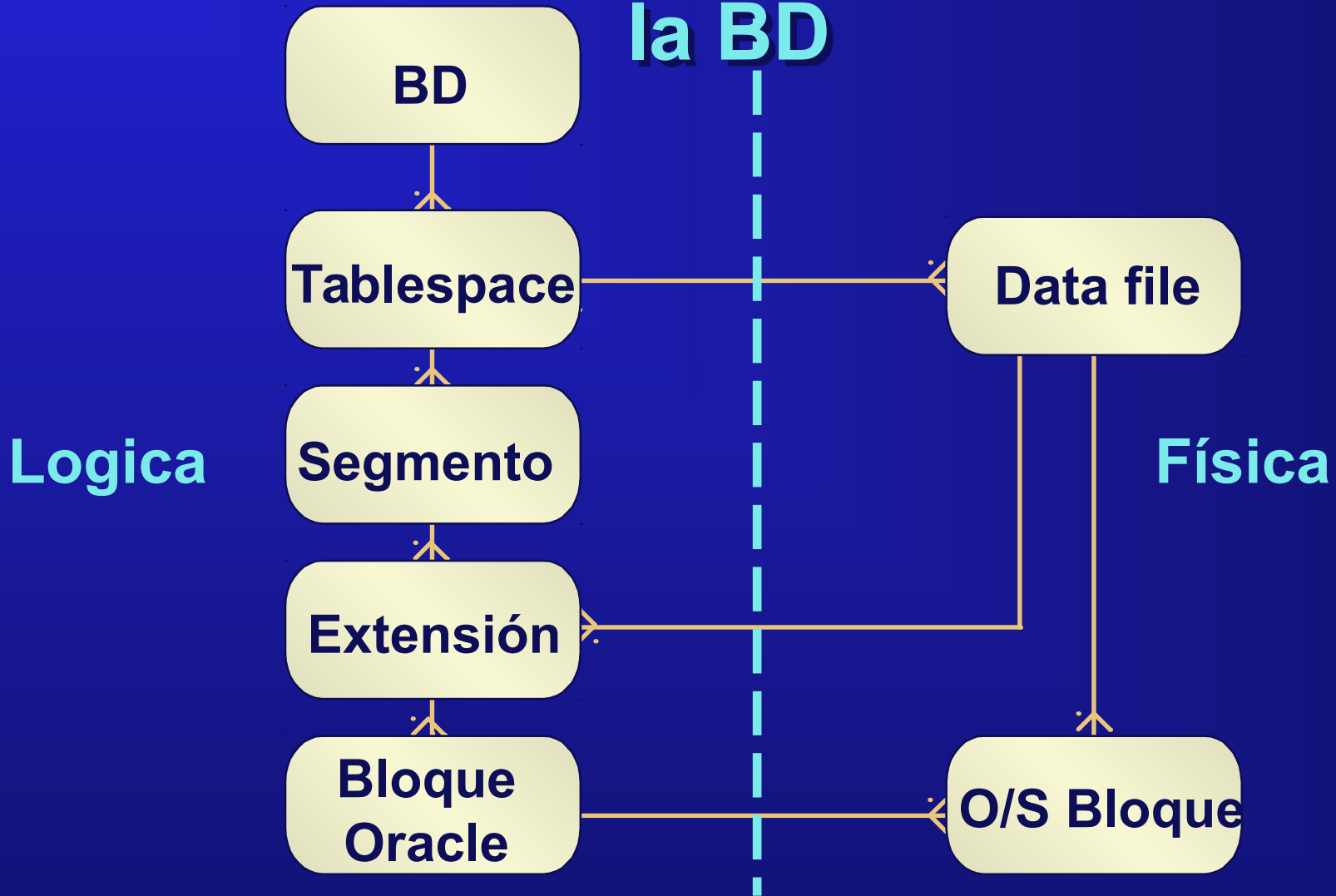
1. **Crear los siguientes tablespaces permanentes:**
 - a) **DATA con extensiones mínimas de 500K, incluyendo la inicial.**
 - b) **RONLY de solo lectura y obligar a que lo sea. Crear una tabla en él.**
2. **Ampliar a 2M el tamaño de DATA01.dbf.**
3. **Crear una tabla en DATA.**
4. **Cambiar el nombre al datafile de DATA;**
5. **Eliminar los tablespaces creados.**

Segmentos, Extensiones y Bloques

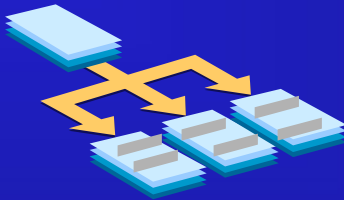
Objetivos

- **Conocer los distintos tipos de segmentos y su uso.**
- **Control del uso de segmentos y extensiones.**
- **Gestión del espacio en bloques ocupado por los objetos.**
- **Información sobre las estructuras de almacenamiento en el catálogo.**
- **Localización de segmentos considerando fragmentación.**

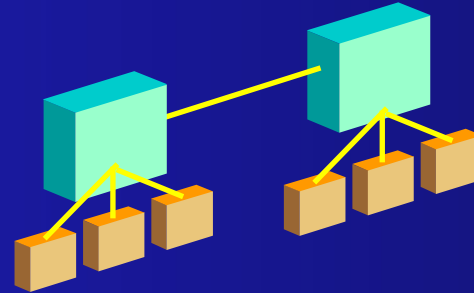
Jerarquía de Almacenamiento de la BD



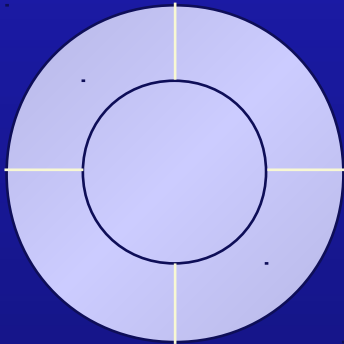
Tipos de Segmentos



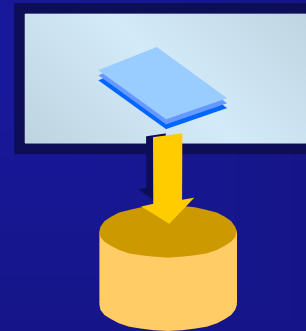
Tabla



**Almacenamiento
de índices**

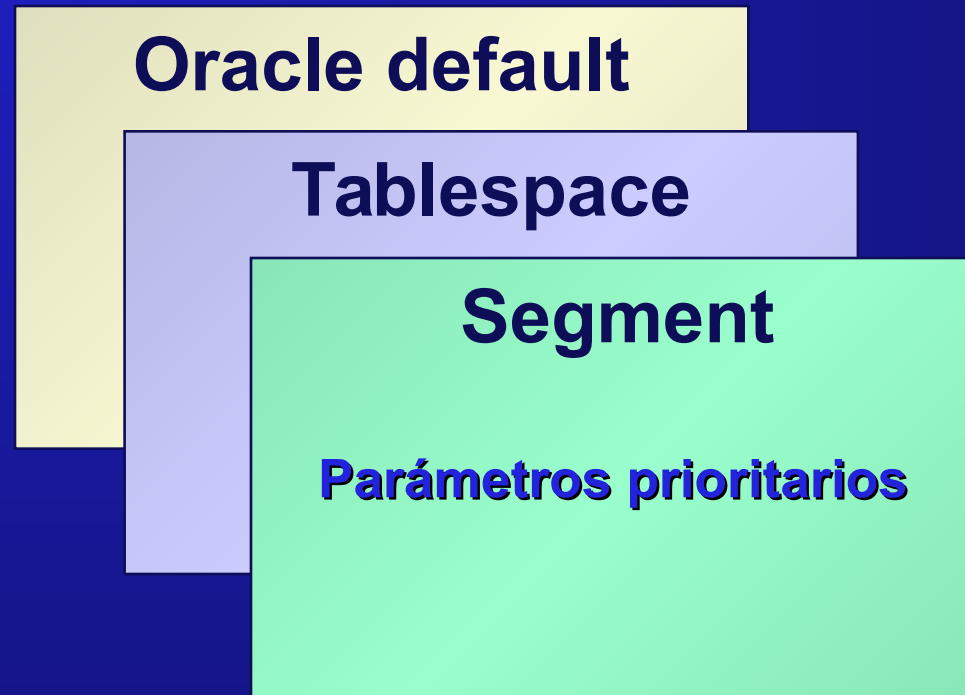


**Segmentos de
Rollback**



**Segmentos
Temporales**

Prioridad de las cláusulas de almacenamiento



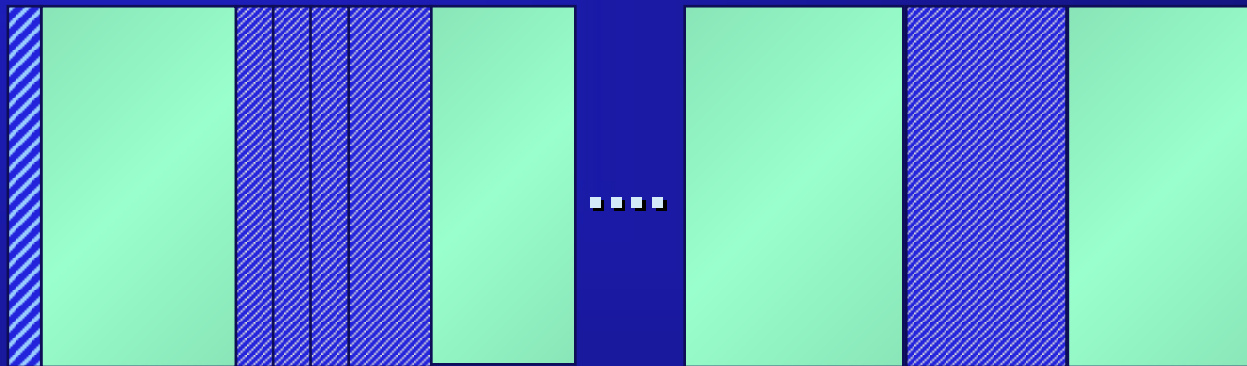
Cualquier parámetro a nivel de segmento anula el correspondiente a nivel de tablespace, excepto MINIMUM EXTENT

Reserva y Liberación de Extensiones

- **Se reserva cuando el segmento es:**
 - + **Creado.**
 - + **Ampliado.**
 - + **Modificado.**
- **Se libera cuando el segmento es:**
 - **Eliminado.**
 - **Modificado.**
 - **Truncado.**

Extensiones Libres y Usadas

Data file



Cabecera



Ext. Usada

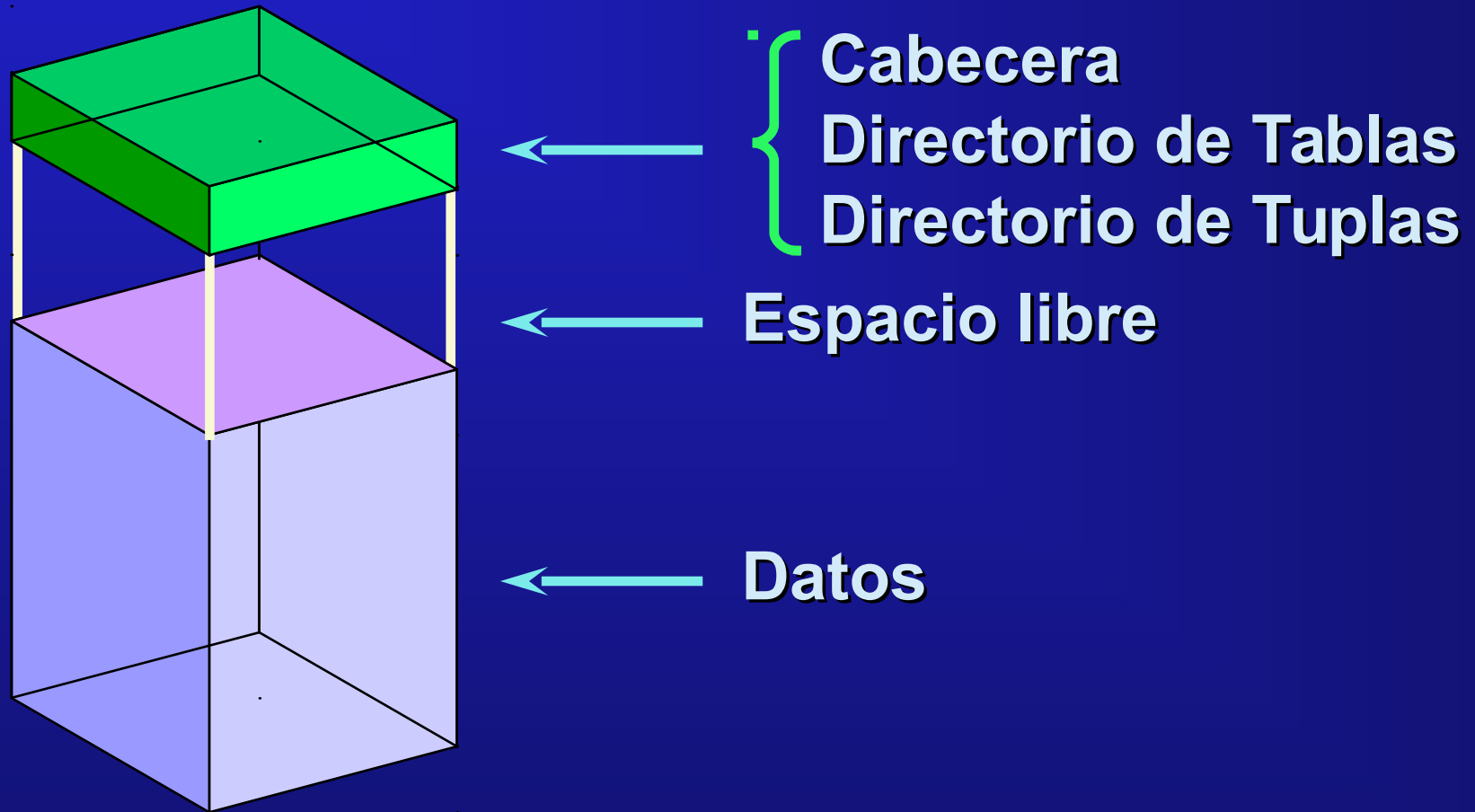


Ext. Libre

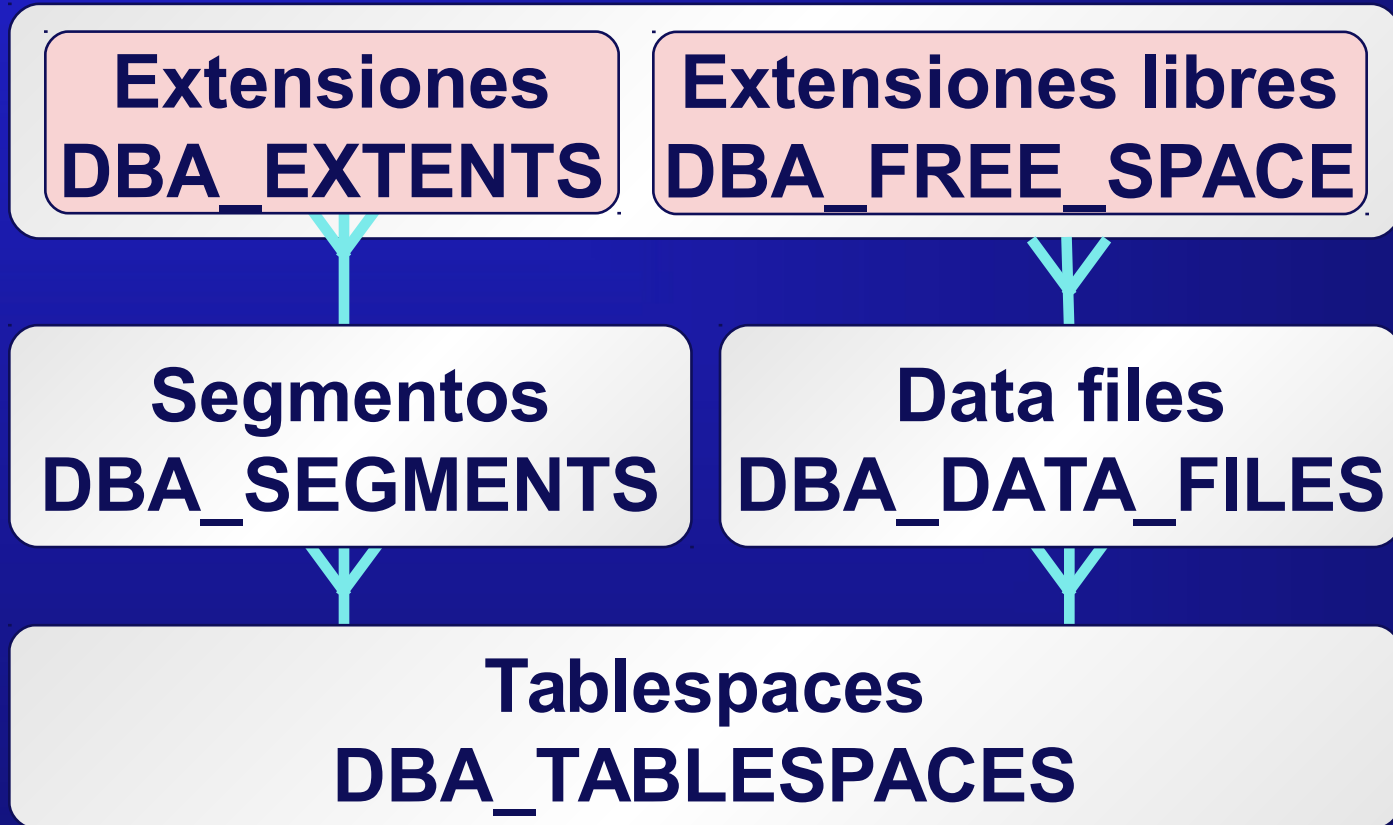
Bloques: Resumen

- **Unidad mínima de E/S.**
- **Compuesto de uno o más bloques de S.O.**
- **Tamaño establecido por el parámetro DB_BLOCK_SIZE.**
- **Se establece al crear la BD.**

Estructura de Bloque



Vistas del Catálogo Relacionadas



Información sobre Segmentos

DBA_SEGMENTS

– Información General

- OWNER
- SEGMENT_NAME
- SEGMENT_TYPE
- TABLESPACE_NAME

– Parámetros

- INITIAL_EXTENT
- NEXT_EXTENT
- MIN_EXTENTS
- MAX_EXTENTS

– Tamaño

- EXTENTS
- BLOCKS

Información sobre Extensiones

DBA_EXTENTS

- **Identificación**
 - OWNER
 - SEGMENT_NAME
 - EXTENT_ID
- **Localización y Tamaño**
 - TABLESPACE_NAME
 - FILE_ID
 - BLOCK_ID
 - BLOCKS

Ejercicios

1. Retocar y ejecutar el script `cr_segs.sql`.
2. Identificar los distintos tipos de segmentos que hay en la BD.
3. Averiguar qué segmentos están a menos de 5 extensiones del límite permitido.
4. ¿Qué ficheros contienen datos de la tabla EMP? (`dba_extents`-`dba_data_files`).

Tablespace UNDO

- Se crea automáticamente un tablespace dedicado a segmentos de rollback (undotbs)
- Parámetro
UNDO_MANAGEMENT=AUTO.
- Parámetro
UNDO_TABLESPACE=undotbs

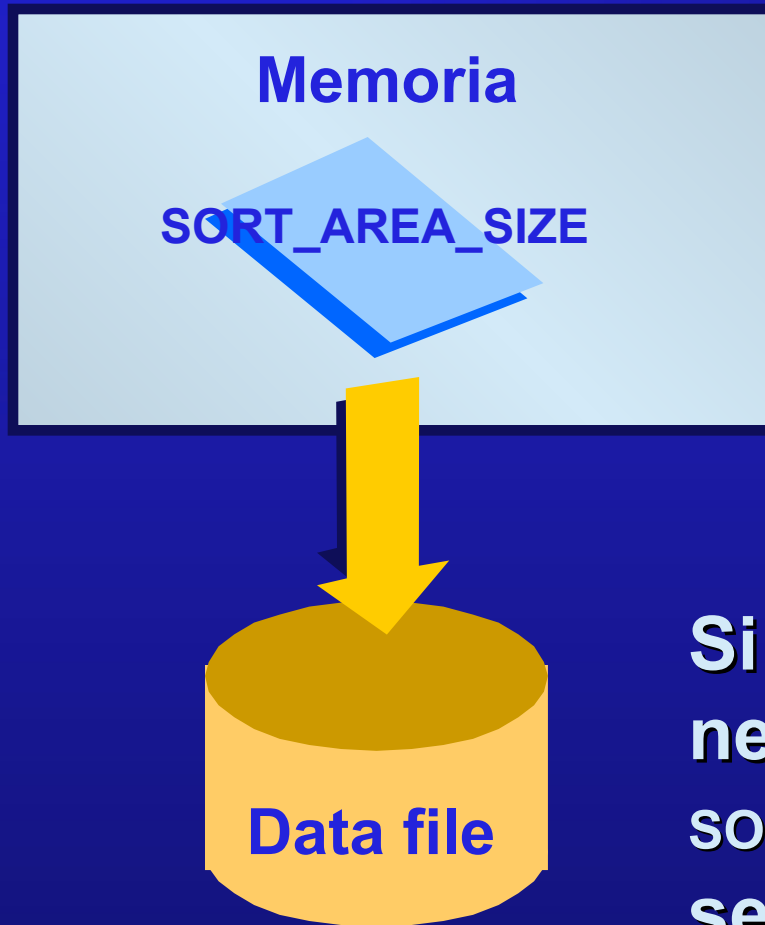
```
CREATE UNDO TABLESPACE UNDOTBS2  
DATAFILE  
'C:\ORACLE\ORADATA\MIBASE\undotbs02' ;
```

Segmentos Temporales

Objetivos

- **Distinguir distintos tipos de segmentos temporales.**
- **Asignación de espacio a segmentos temporales.**
- **Información sobre segmentos temporales en el catálogo.**

Segmento Temporal



- SELECT... ORDER BY
- CREATE INDEX
- SELECT DISTINCT
- SELECT... GROUP BY
- SELECT... UNION

Si la operación a realizar necesita espacio adicional al SORT AREA SIZE, lo toma de los segmentos temporales

Segmentos Temporales en Tablespaces Temporales

- Un tablespace temporal (temporary) sólo puede tener segmentos temporales

```
CREATE TABLESPACE <tb_id> TEMPORARY  
DATAFILE <file_id> ....
```

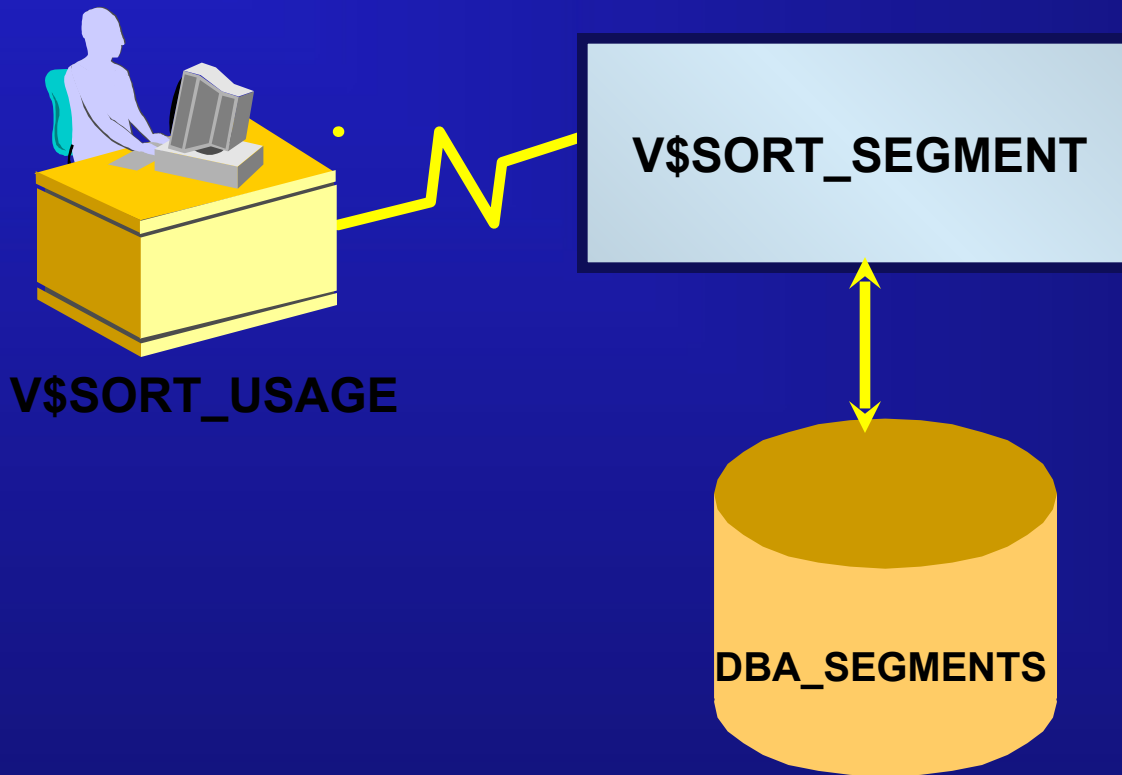
- Un tablespace puede cambiar de temporal a permanente y viceversa

```
ALTER TABLESPACE <tb_id> TEMPORARY | PERMANENT;
```

Segmentos Temporales en Tablespaces Temporales

- **Es suficiente con un segmento por tablespace.**
- **Se crea cuando se ejecuta la primera operación que requiere ordenar.**
- **Se reutiliza el espacio continuamente.**
- **Se libera el segmento tras shutdown.**
- **No hay limitación del número de extensiones.**

Información sobre Segmentos Temporales



Estadísticas de Segmentos Temporales

V\$SORT_SEGMENT

- TABLESPACE_NAME
- EXTENT_SIZE
- TOTAL_EXTENTS
- TOTAL_BLOCKS
- USED_EXTENTS
- USED_BLOCKS
- FREE_EXTENTS
- FREE_BLOCKS
- MAX_SORT_SIZE
- MAX_SORT_BLOCKS

Segmentos Temporales en Uso

