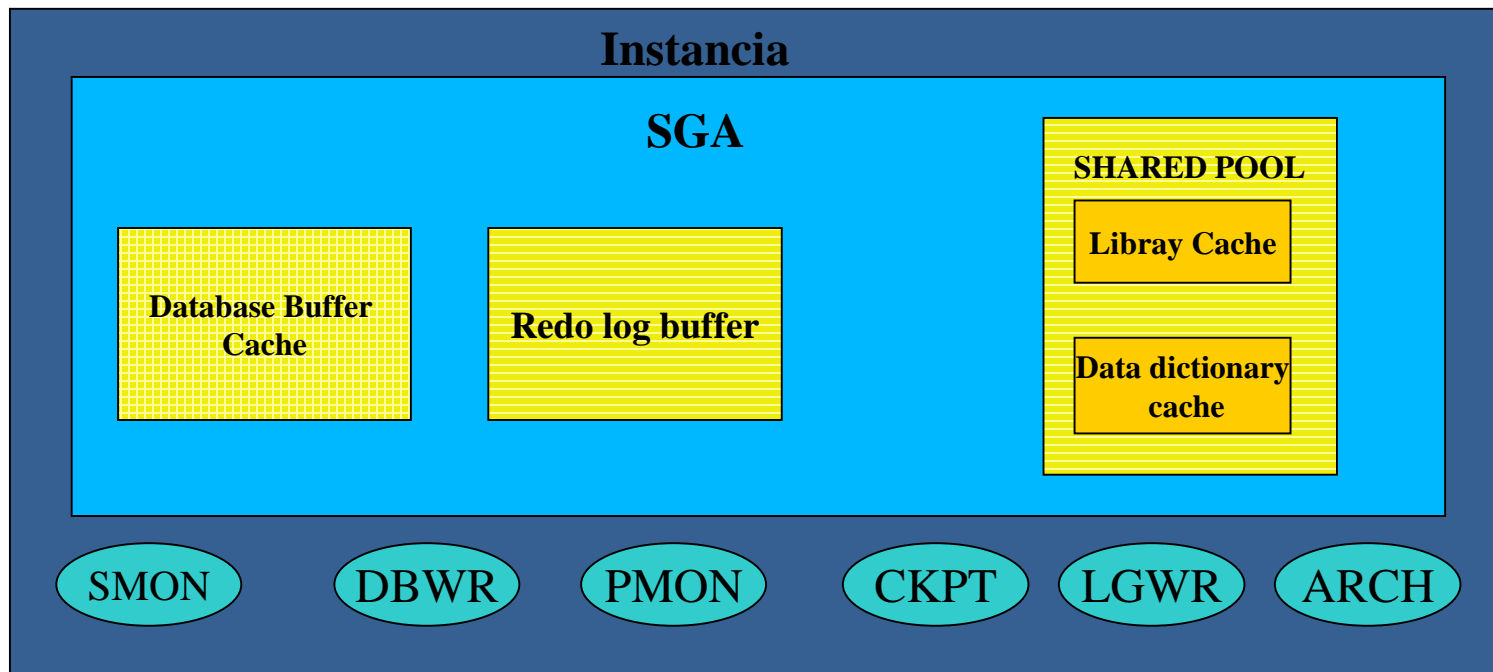




ARQUITECTURA DE LA BD ORACLE

LA INSTANCIA DE ORACLE

- El Oracle Server consiste de la base de Datos y la instancia Oracle.
- Una instancia consiste de una estructura de memoria llamada System Global Area (SGA), y de procesos que operan en background utilizados para administrar la Base de Datos.
- Una instancia de Oracle siempre se corresponde con una sola base de datos.



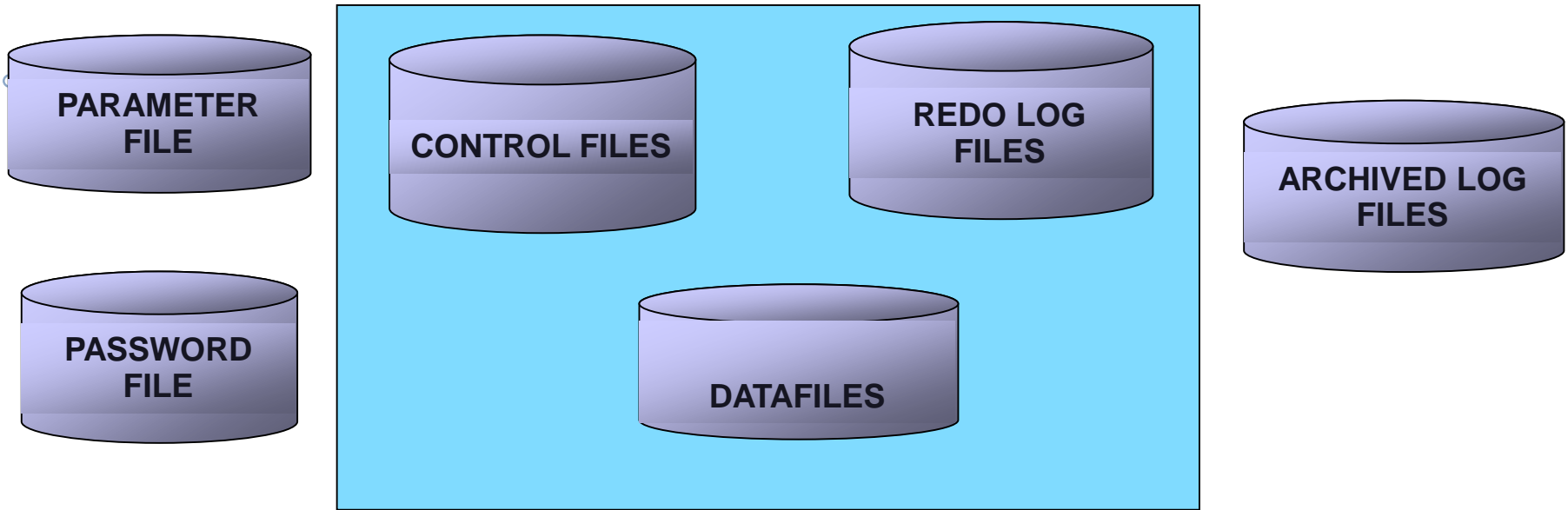
INSTANCIA DE ORACLE: Procesos de background

- **SMON (System Monitor):** Es el supervisor del sistema y se encarga de todas las recuperaciones que sean necesarias durante el arranque.
- **DBWR (Database Writer):** El proceso DBWR es el responsable de gestionar el contenido de los buffers de datos y del caché del diccionario.
- **PMON (Process Monitor):** Este proceso restaura las transacciones no validadas de los procesos de usuario que abortan, liberando los bloqueos y los recursos de la SGA.
- **CKPT (Checkpoint):** Este proceso escribe en los ficheros de control los checkpoints. Estos puntos de sincronización son referencias al estado coherente de todos los ficheros de la BD en un instante determinado, en un punto de sincronización.
- **LGWR (Log Writer):** Este proceso es que se encarga de escribir el contenido de los buffers del redo log en el disco.
- **ARCH (Proceso archivador):** El proceso archivador tiene que ver con los ficheros redo log. Por defecto, estos ficheros se reutilizan de manera cíclica de modo que se van perdiendo los registros redo log que tienen una cierta antigüedad.

System Global Area SGA

- El SGA es una región de memoria que contiene datos e información de control para el Server.
- El SGA se asigna en la memoria virtual del equipo donde reside el Oracle y comprende las siguientes estructuras de memoria:
 - **Shared Pool:** se usa para almacenar información tal como la sentencia SQL más recientemente utilizada y los datos del diccionario más recientemente utilizados
 - **Database buffer cache:** usado para almacenar los datos usados más recientemente
 - **Redo log buffer:** usado para registrar los cambios hechos a la base de datos usando la instancia

Base de Datos



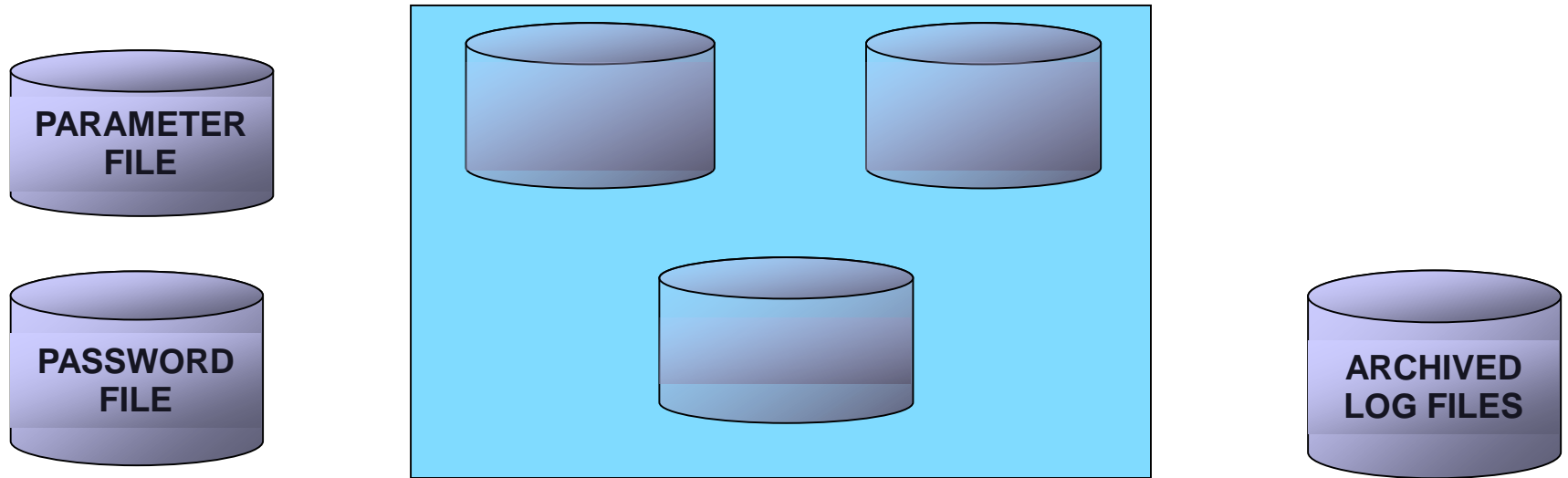
- **La Arquitectura de la BD ORACLE comprende una serie de archivos requeridos para su funcionamiento:**
 - Archivos de datos : Archivos de control, Archivos de datos y archivos de log de recuperación
 - Otros archivos importantes son: Archivo de parámetro, Archivo de passwords y archivo de log de recuperación off line



Archivos de Datos

- **CONTROL FILES:** Contienen la información necesaria para mantener y verificar la integridad de la base de datos.
- **REDO LOG FILES:** Contienen un registro de los cambios realizados en la base de datos para asegurar la reconstrucción de los datos en caso de fallas.
- **DATA FILES:** Almacena el diccionario de datos, los objetos del usuario, y los datos.

Base de Datos



- **Parameter File:** usado para definir las características de la instancia
nls_language, cpu_count, db_block_size
- **Password File:** Usado para autenticar los usuarios privilegiados de la BD
- **Archived redo log:** copias offline de los “redo log” files que podrían utilizarse para recuperar los datos cuando falla el medio físico



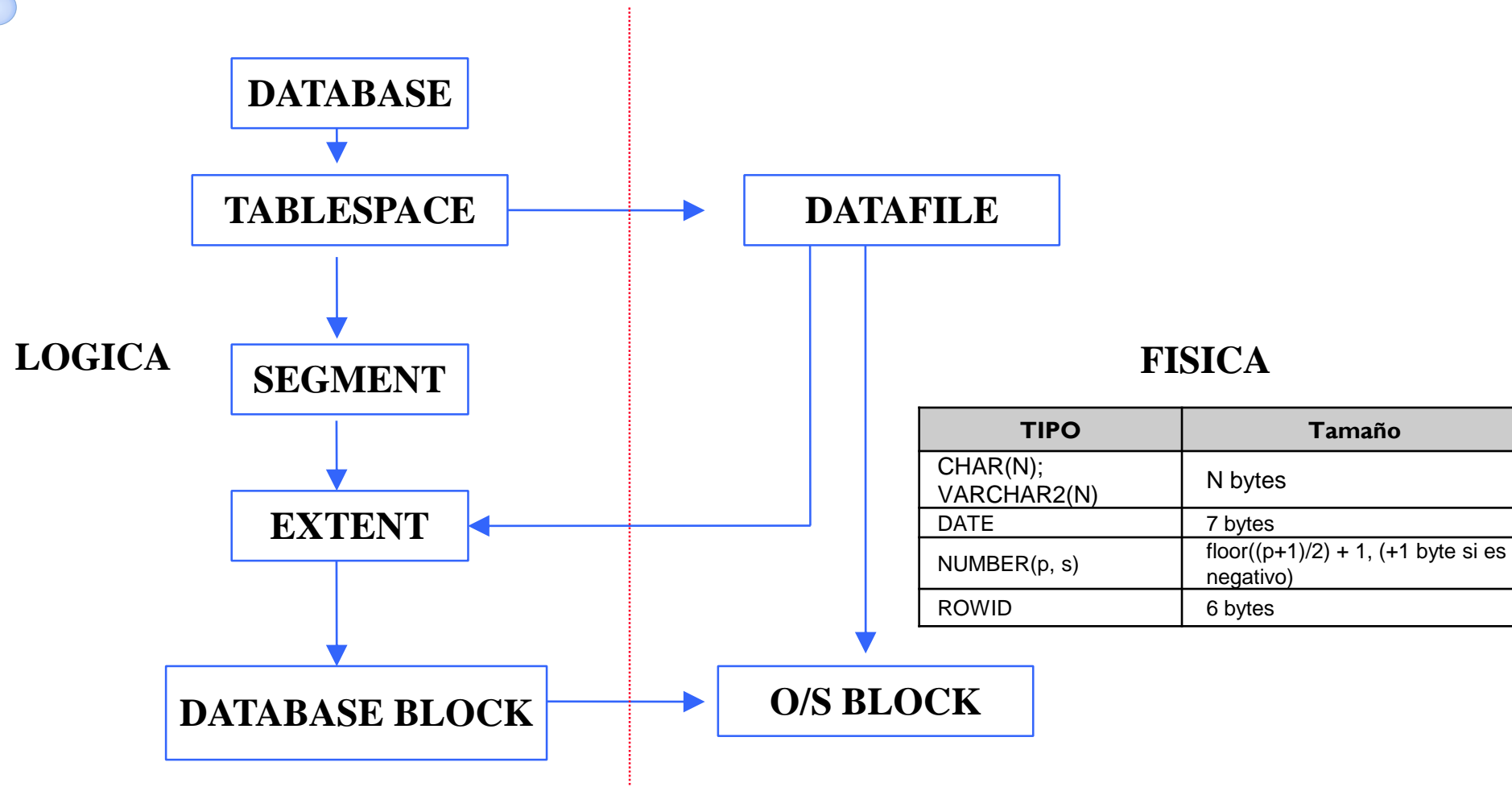
Tablespaces

- El tablespace es una estructura lógica creada y conocida por la Base de Datos ORACLE. Consiste de archivos de datos y archivos temporales
- Existen tablespaces de datos, temporales y tablespaces para operaciones UNDO.

Estructura lógica de la base de datos

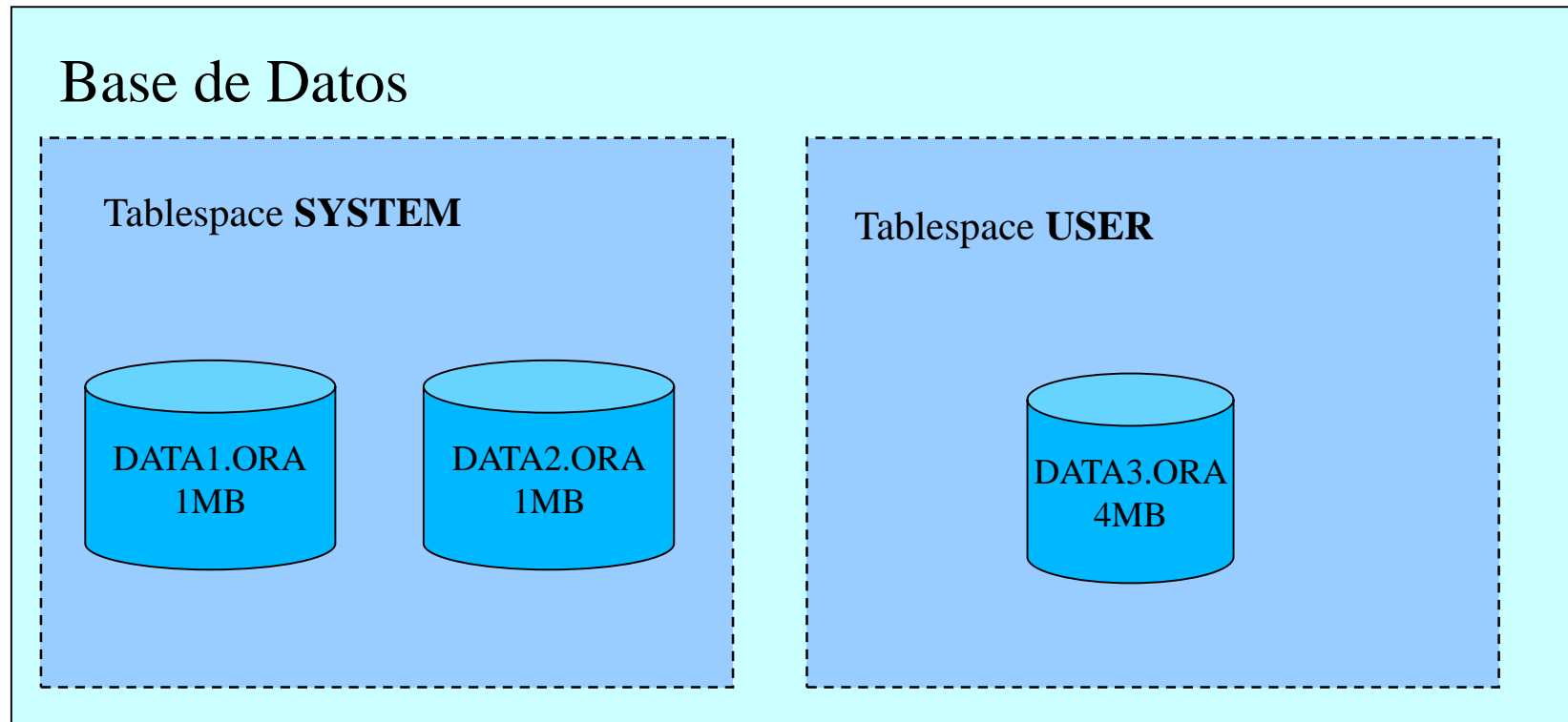
- Cada Base de Datos está lógicamente dividida en uno o más tablespaces, pero cada tablespace puede pertenecer solamente a una BD
- Cada tablespace consiste de 1 o más archivos de sistema (datafiles) que físicamente almacenan la estructura lógica mencionada.
- El tamaño combinado de todos los datafiles que conforman un tablespace es la capacidad de almacenamiento del tablespace, y la suma de estos últimos es la capacidad de almacenamiento de la BD.
- A excepción del tablespace SYSTEM o un tablespace con un rollback segment activo, los tablespaces pueden ser puestos off line teniendo la BD en actividad
- Los tablespaces pueden también de lectura-escritura o sólo de lectura

Estructura lógica de la base de datos

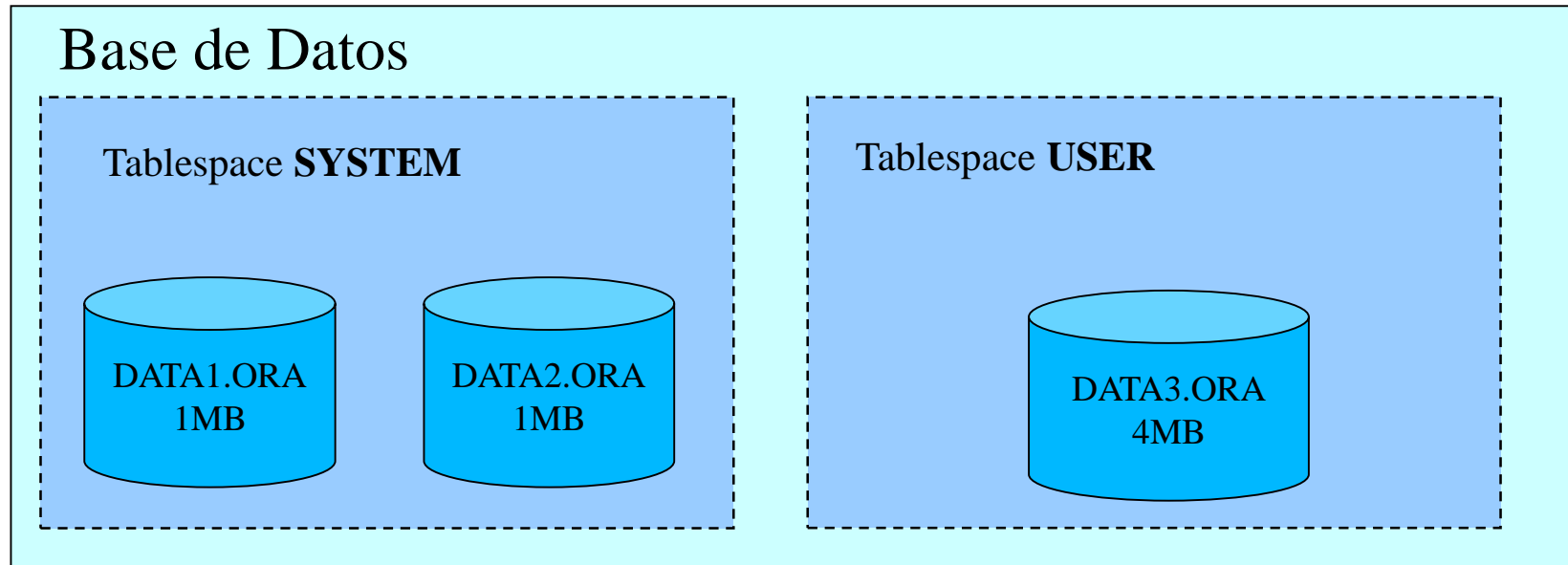


Estructura lógica de la base de datos

Los **tablespaces**: son unidades lógicas de almacenamiento, que agrupan a su vez estructuras lógicas relacionadas.



Estructura lógica de la base de datos



- **2 Tablespaces:**
 - System: 2 Datafiles → Tamaño total: 2MB
 - User: 1 Datafile → Tamaño total: 4MB
- Capacidad total de la BD: 6MB

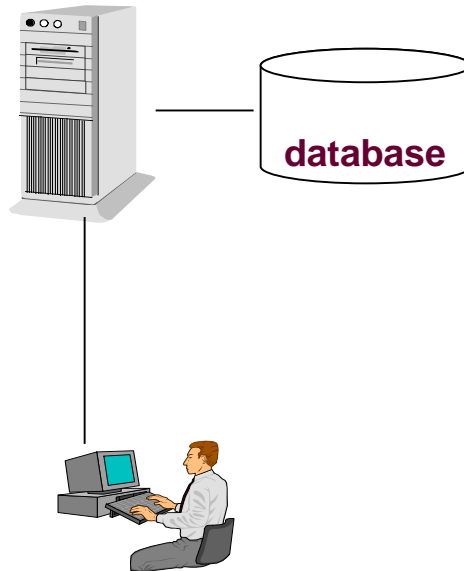
Relación entre segments, extents y Data Blocks

- Un **segmento** es el espacio reservado para un tipo específico de estructura lógica de almacenamiento dentro de un tablespace. Ejemplos:
 - Segmentos de tablas
 - Segmentos de índices
 - Segmentos temporales
 - Segmentos de rollback
- **Extents:** Es el siguiente nivel lógico del espacio de la BD. Un extent es un conjunto de n° de bloques continuos. Un extent no puede distribuirse en más de un archivo físico
- **Data Blocks:** El el nivel más pequeño de almacenamiento. Un data block se corresponde con 1 o más bloques de datos físicos. Se indica en el parámetro DB_BLOCK_SIZE

Administrar el espacio de la base de datos

**Servidor
Dominio
Principal
y de Base
de Datos**

BASE DE DATOS:



TABLESPACES

SYSTEM

DATA

TEMP

ROLLBACK

CONTENIDO

- **Diccionario de Datos**
- **Objetos creados por el**
- **Servidor de Base de Datos**

- **Objetos creados por el**
- **usuario : Ej.**
- **B_EMPLEADOS**

- **Objetos creados**
- **temporalmente por**
- **el S.B.D.**

- **Segmentos de**
- **Rollback.**
- **Paraprocetamiento de**
- **datos.**

Creación de Tablespace: Sintaxis

```
CREATE TABLESPACE tablespace  
    DATAFILE 'especificación del archivo fisico'  
    [DEFAULT STORAGE (clausula storage)]  
    [PERMANENT | TEMPORARY]  
    [ONLINE | OFFLINE] ;
```

- **Se deben tener privilegios específicos:**
 - **CREATE TABLESPACE**
 - **Generalmente rol de Administrador de Base de Datos (DBA).**

Creación de Tablespace: Ejemplo

```
SQL> CREATE TABLESPACE BASED2
2  DATAFILE '/ORANT/ORADATA/BASED2_01.dbf' SIZE 50M
3  DEFAULT STORAGE (INITIAL 10K NEXT 50K
4                      MAXEXTENTS 100
5                      PCTINCREASE 0)
6  ONLINE;
```


Ampliando un tablespace: Ejemplo

```
SQL> ALTER TABLESPACE BASED2  
2 ADD DATAFILE '/ORANT/ORADATA/BASED2_02.dbf' SIZE 50M;
```

Poniendo un tablespace fuera de línea

```
SQL> ALTER TABLESPACE BASED2 OFFLINE NORMAL;
```

Eliminación de un tablespace

```
SQL> DROP TABLESPACE <espacio de tablas>  
2  [INCLUDING CONTENTS [CASCADE CONSTRAINTS]];
```

- **INCLUDING CONTENTS:** Elimina todos los segmentos en el TABLESPACE
- **CASCADE CONSTRAINTS:** Elimina constraints de integridad referencial de las tablas fuera del tablespace que se refieren a las PK de las tablas del tablespace eliminado

Observaciones:

- Si el tablespace contiene aún datos se requiere la opción INCLUDING CONTENTS
- Se recomienda que el tablespace se ponga OFFLINE antes de eliminarlo, para asegurarse que ninguna transacción accede a los segmentos del mismo.



*Espacio de tabla **TEMPORARY***

- Los tablespaces temporales contienen objetos del esquema que se generan solamente durante la duración de la sesión. Los objetos en los “temporary tablespaces” se almacenan también en archivos temporales (tempfiles) .
- Se especifica indicando la opción 'TEMPORARY' en la creación del tablespace.

Espacio de tabla TEMPORARY. Ejemplo

El siguiente ejemplo determina el directorio por default para la creación de los datafiles:

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST =  
' $ORACLE_HOME/rdbms/dbs ' ;
```

Posteriormente los tablespaces se generarán en el destino definido:

```
CREATE TEMPORARY TABLESPACE temp_demo TEMPFILE  
'temp01.dbf' SIZE 5M AUTOEXTEND ON;
```



Espacio de tabla de ROLLBACK

- **Un segmento de ROLLBACK es usado para mantener los valores anteriores cuando una transacción está realizando cambios en la base de datos.**
- **Por tanto, almacena bloques de información tales como el archivo, el id del bloque y los datos con los valores anteriores.**
- **Varias transacciones concurrentes pueden usar un solo rollback segment.**



Creación de un segmento de rollback

```
CREATE [PUBLIC] ROLLBACK SEGMENT  
    <nombre>  
[TABLESPACE nombre]  
[STORAGE (clausula de storage)]:
```

Creación de un segmento de rollback

```
CREATE ROLLBACK SEGMENT rbs01  
TABLESPACE rbs  
STORAGE (INITIAL 100K NEXT 100K OPTIMAL 4M  
MINEXTENTS 20 MAXEXTENTS 100);
```

Otros Ejemplos

- Se habilita el rollback segment

ALTER ROLLBACK SEGMENT RB_TEMP ONLINE;

- Cambio de asignación de tablespace por defecto y tablespace temporal para el usuario SYSTEM.

**ALTER USER SYSTEM TEMPORARY TABLESPACE
TEMPORARY_DATA;**

ALTER USER SYSTEM DEFAULT TABLESPACE USER_DATA;

- Creación de rollback segment público (Estimar la cantidad de segmentos de rollback de acuerdo a los procesos ejecutados y a los usuarios que son conectados. Se recomienda un segmento de rollback por cada 4 (cuatro) usuarios conectados)

**CREATE PUBLIC ROLLBACK SEGMENT RB1 STORAGE(INITIAL
50K NEXT 50K)
TABLESPACE ROLLBACK_DATA;**