



SC-503

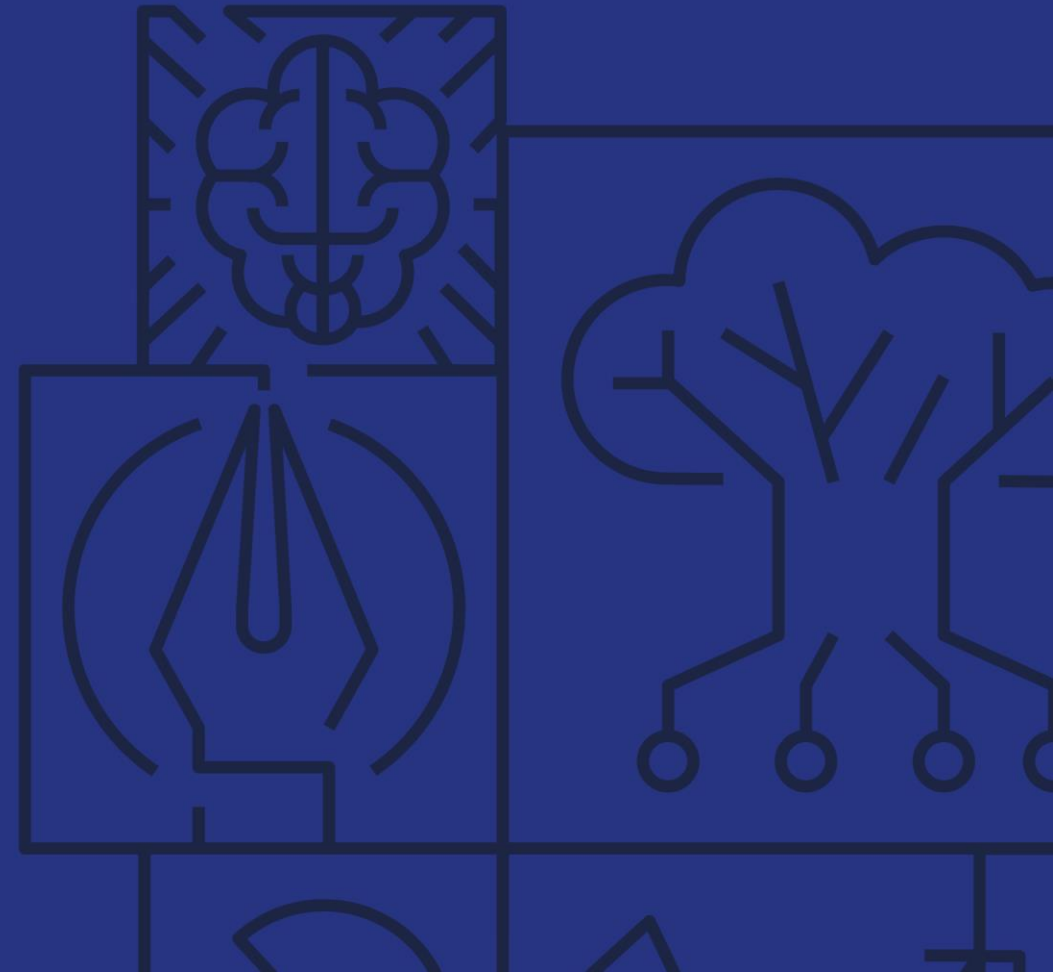
Administración de Base de Datos





SEMANA 2

ARQUITECTURA



Arquitectura - Conceptos Básicos

- El servidor Oracle funciona en una arquitectura cliente – servidor
 - Cliente
 - El cliente corre los programas como por ejemplo, SQL*Plus, .NET, o algún otro programa que acceda a la información de la base de datos e interactúe de alguna manera con alguna entrada.
 - Servidor
 - Corre el Software de Base de datos de Oracle y maneja todas las funciones de concurrencia y datos compartidos.



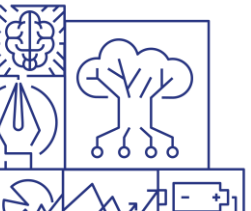
Arquitectura - Conceptos Básicos

- ORACLE es un sistema gestor de bases de datos relacionales. Sus dos principales componentes son:
 - La instancia
 - La base de datos



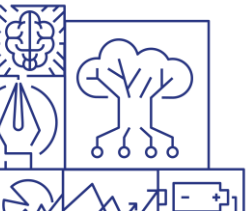
Database vs Instance

- ¿Qué es una base de datos ORACLE?
 - Es el set de archivos físicos en donde se guarda la información tanto de las aplicaciones como la meta data.
- ¿Qué es una instancia de ORACLE?
 - Es el software (y la memoria) que ORACLE usa para manipular los datos en la base de datos
- Una base de datos ORACLE puede ser abierta por más de una instancia de ORACLE
- Una instancia de ORACLE solo puede abrir una base de datos ORACLE



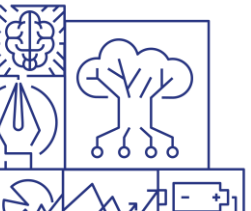
Oracle Instance

- Cuando la instancia arranca esta hace dos cosas:
 - Crea el SGA (SystemGlobalArea) en memoria
 - Inicia uno o más de losbackgroundprocesses



Instance– SGA

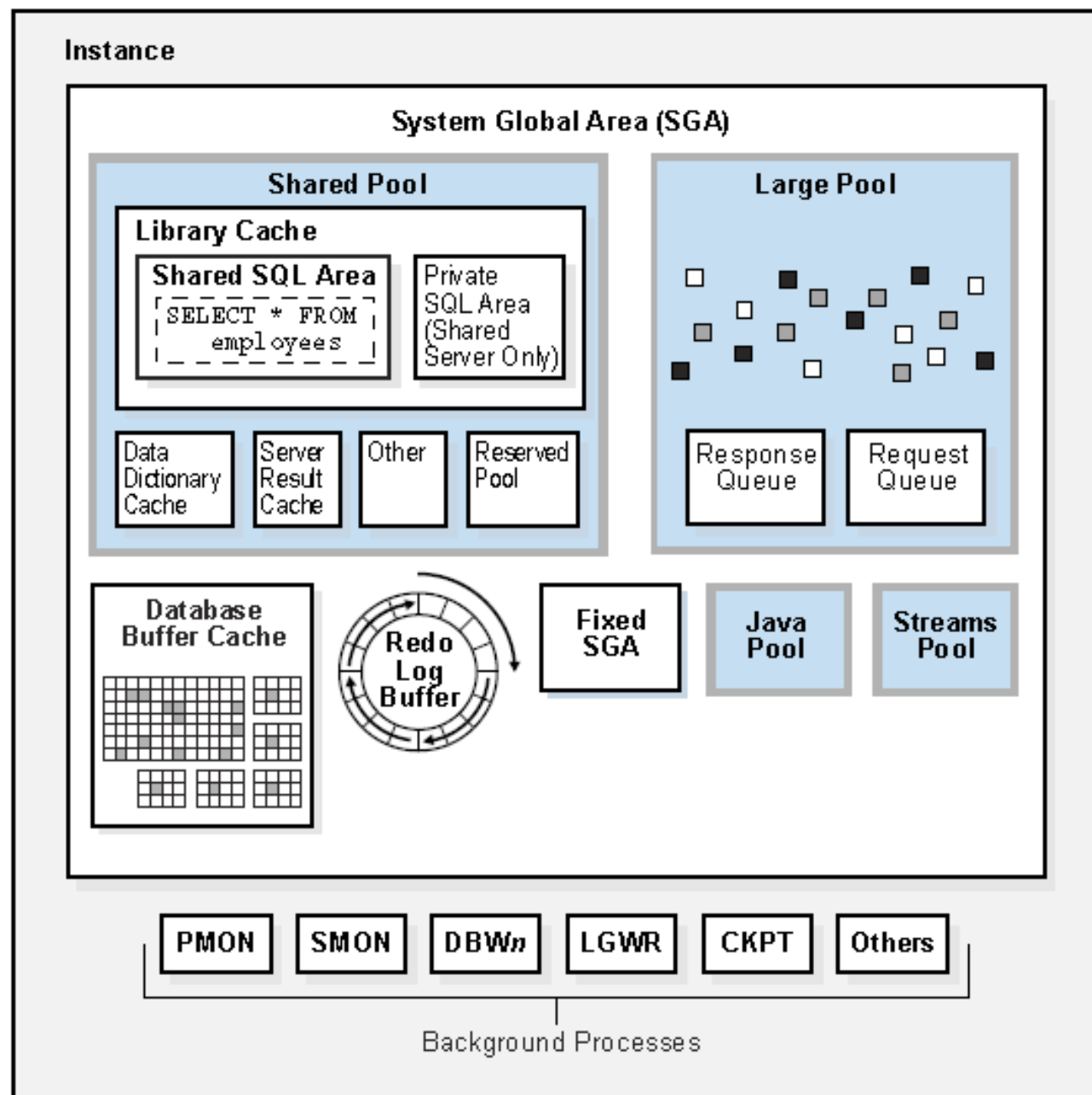
- El SGA sirve muchos propósitos entre los cuales están:
 - Mantener las estructuras internas que son accedidas por los procesos y threads simultáneamente
 - Cache los bloques de datos que fueron leídos
 - Guardar la información de redo antes de ser escrita en los archivos de redo online
 - Guardar la información de SQL execution plans



Instance– SGA

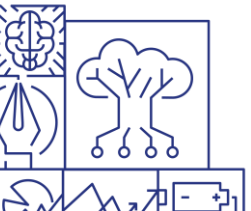
- El SGA es compartido por los procesos de Oracle, estos incluyen:
 - Los procesos del servidor (service handlers)
 - Los background processes
- Nota: Los procesos de Oracle no forman parte del SGA, estos residen en porciones separadas de la memoria





Estructura de la Memoria

- Los procesos del usuario (cliente) y del servidor se comunican consigo mismos y entre sí a través de estructuras de memoria.
- Oracle utiliza dos tipos de estructuras de memoria:
 - **Área global del programa (PGA):** es la zona de memoria usada por un único proceso de usuario de Oracle, conteniendo datos e información del proceso.
 - **Área global del sistema (SGA):** permite la comunicación entre los procesos cliente – servidor. También mantiene la información más consultada.



Instance– SGA

- El SGA consiste de varias secciones de memoria que cumplen distintos propósitos
- Cada una de estas secciones consiste de uno o más **granules** de memoria.
 - Un granule es una unidad de memoria adyacente, normalmente en Oracle de 4MB o 16MB dependiendo del tamaño del SGA
- Note que la información del SGA puede ser vista con la vista V\$SGASTAT



Instance– SGA

- Los componentes más importantes del SGA son:
 - DatabaseBuffer Cache
 - Redo Log Buffer
 - SharedPool
 - LargePool
 - Java Pool
 - StreamsPool
 - FixedSGA

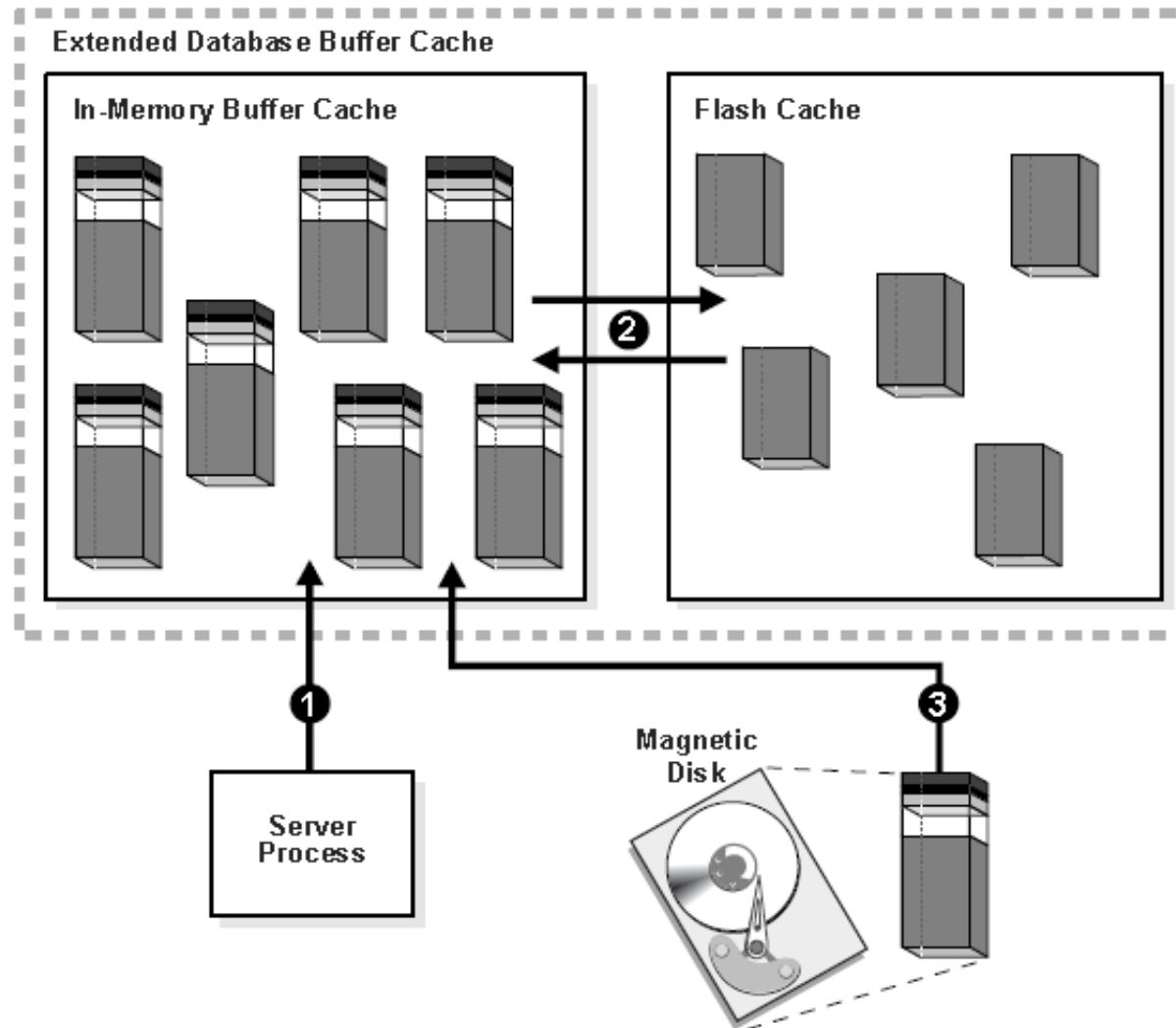


Database Buffer Cache

- Guarda copias de los bloques de datos leídos de los archivos
- Objetivos de este Cache
 - Optimizar el I/O físico
 - La base de datos actualiza los bloques de datos en el cache y guarda los cambios en el redo buffer, cuando se hace COMMIT, la base de datos escribe los redo buffers, pero no escribe los datos inmediatamente, esto lo hace el database writer (DBWn) en el background.
 - Mantener los bloques que se usan frecuentemente en el cache y escribir al disco los que no usan
 - Database Smart Flash Cache (flash cache)



Database Buffer Cache



Buffer Pools

- Buffer Pools
 - Default Pool, donde los datos son normalmente escritos
 - KeepPool, es para mantener bloques que se usan frecuentemente pero que ya no se pueden mantener en el default pool, porque son viejos o por falta de espacio
 - RecyclePool, se mantienen bloques que no se usan frecuentemente, esto es para no guardar bloques innecesariamente en el cache
 - El tamaño del bloque es de 8K, pero hay pools especiales para bloques con otros tamaños, de 2K, 4K y 16K



Shared Pool

Esta estructura tiene la finalidad de guardar la definición de los objetos que son reutilizados y compartidos por los diversos usuarios.

El parámetro de configuración de Oracle que define el tamaño de la estructura es **SHARED_POOL_SIZE**.

Se encuentra dividido en dos áreas de memoria:

- Data dictionary cache.
- Library cache.



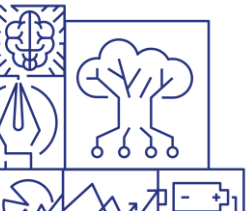
Shared Pool

Data Dictionary Cache

- Es una memoria encargada de guardar información acerca de la estructura de los objetos accesados.

Library Cache

- Guarda las definiciones de las sentencias SQL y PLSQL ejecutadas (Planes de Ejecución).
- La ejecución de una sentencia SQL tiene tres etapas:
 1. Parsing
 2. Ejecución
 3. Fetch

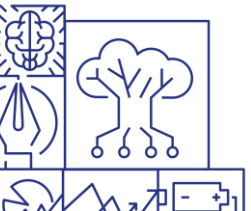


Redo log buffer

Esta memoria tiene como finalidad guardar las operaciones que se realizan sobre los bloques de los archivos de datos. Se guarda información que permita rehacer instrucciones del tipo delete, insert, update, drop, alter, create; lo cual permite ser un apoyo en caso de ser necesaria una recuperación de la base de datos.

Es un buffer circular.

El parámetro de configuración de Oracle que define el tamaño de la cache es **LOG_BUFFER**.



Large pool

Esta estructura de memoria es opcional, su configuración depende del modo en que se configure el servidor Oracle.

En un ambiente dedicado posiblemente no sea necesaria su utilización a menos que se pretendan utilizar las nuevas utilidades de backup y recuperación como RMAN (utilidad incluida en ultimas versiones).

El parámetro de configuración de Oracle que define el tamaño de la estructura es `LARGE_POOL_SIZE`.



Java pool

Otra estructura de memoria opcional que se configura cuando incluimos código java en nuestra base de datos.

Busca guardar información acerca de la interpretación de las sentencias java que se ejecutan.

El parámetro de configuración de Oracle que define el tamaño de la estructura es `JAVA_POOL_SIZE`.



Procesos de Usuario

Cuando un usuario se conecta a la base de datos, se crea un proceso de usuario que se encarga de ejecutar el código de aplicación del usuario y manejar el perfil del usuario con sus variables de ambiente.

Los procesos de usuario no se pueden comunicar directamente con la base de datos, únicamente lo hacen a través de procesos servidores.



Procesos de Servidor

Ejecutan las órdenes SQL de los usuarios y llevan los datos al “database buffer cache”, para que los procesos del usuario puedan tener acceso a los datos.

Se pueden tener distintas arquitecturas para trabajar en ORACLE, según los tipos de servidores:

- Dedicados
- Compartidos (Multihilos).



Procesos de Segundo Plano (Background)

Se encargan de traer datos desde y hacia la SGA.

Mejora el desempeño al consolidar las tareas que son impartidas por todos los usuarios.

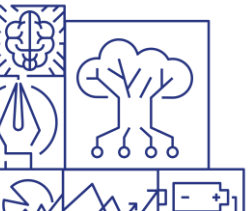
Cada proceso de base tiene su propia área de memoria.



Procesos de Segundo Plano (Background)

Process Monitor.

Actúa cuando falla un proceso de usuario, liberando los recursos que tuviera asignados (memoria, entre otros) y deshaciendo los cambios que hubiese realizado desde su último **COMMIT**.

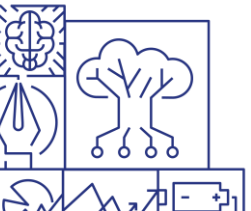


Procesos de Segundo Plano (Background)

System Monitor.

Maneja la recuperación de la BD a partir del fallo de una Instancia (esto es, cuando las estructuras de memoria y los procesos que componen la Instancia no pueden continuar por algún motivo).

Chequea periódicamente los espacios de disco para determinar si una pequeños fragmentos de espacios libres.



Procesos de Segundo Plano (Background)

Escribe los bloques de datos (y rollback) de la SGA (data buffer cache) en los DataFiles. Esto lo hace de forma asíncrona, cuando:

- Sucede un checkpoint.
- El número de buffers modificados alcanza un umbral.
- No quedan buffers libres.
- Ocurre un timeout.
- Ponemos un tablespace offline.
- Dejamos un tablespace en modo readonly.
- Borramos o “truncamos” una tabla.
- `ALTER TABLESPACE nombretsp BEGIN BACKUP.`



Procesos de Segundo Plano (Background)

Log writer.

Es el proceso que escribe datos desde la SGA a los archivos de rehacer (redo log files) que sirven en caso de fallas en la instancia.

Realiza escrituras secuenciales del contenido de la redo log buffer cache en los ficheros redo log. ¿Cuándo?

- Se hace commit.
- La redo log buffer cache se llena 1/3.
- Hay 1Mb de cambios en la redo log buffer cache.
- Como mucho, cada 3 segundos.
- Siempre antes que escriba el DBWR.

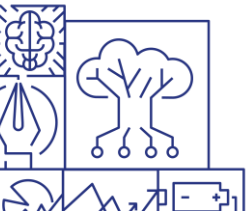


Procesos de Segundo Plano (Background)

CheckPoint (punto de comprobación).

Es un proceso opcional que ocurre cuando los usuarios conectados a la base de datos, hacen solicitudes de reconocimientos de datos.

Asegura que el DBWR escriba los datos en disco de un bloque de datos modificado en memoria.

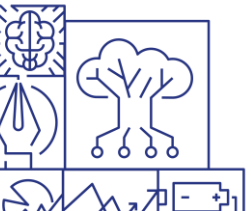


Procesos de Segundo Plano (Background)

Recoverer (RECO): Es un proceso de *background* opcional que recupera ante las fallas, en una transacción en ambientes distribuidos.

Archiver (ARCH): Es opcional. Copia los ficheros del *Redo Log* a un dispositivo predeterminado cuando éste está lleno. Este proceso solamente se presenta cuando el *Redo Log* se usa en modo **ARCHIVELOG** y el archivo automático está activado.

Lock (LCKn): Opcional. Solamente para servidores paralelos. Gestiona los bloqueos entre distintas Instancias.

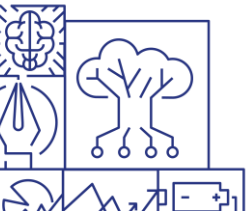


Diccionario de Datos

Es un conjunto de tablas de sólo lectura con los metadatos (son datos que describen otros datos).

En general, un grupo de metadatos se refiere a un grupo de datos, llamado *recurso*.

El concepto de metadatos es análogo al uso de índices para localizar objetos en vez de datos. Por ejemplo, en una biblioteca se usan fichas que especifican autores, títulos, casas editoriales y lugares para buscar libros. Así, los metadatos ayudan a ubicar datos).



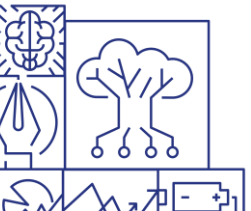
Diccionario de Datos

La definición de todos los objetos o estructuras internas del esquema (tablas, vistas, índices, clusters, sinónimos, secuencias, procedimientos, paquetes, funciones, triggers, entre otros).

Cuánto espacio está reservado o tienen ocupado los objetos del esquema.

Los valores por defecto de las columnas.

Información sobre las restricciones de integridad.



Diccionario de Datos

- Nombres de los usuarios de Oracle.
- Privilegios y Roles que los usuarios tienen concedidos.
- Información sobre auditoría de la BD.
- Otras informaciones de carácter general (estadísticas, entre otras).



Diccionario de Datos

Tablas:

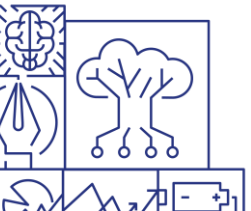
- Son propiedad del usuario **SYS**.
- Esas tablas no son muy útiles directamente porque están en un formato muy críptico.



Diccionario de Datos

Tablas:

- La Tabla **DUAL**: Es una pequeña tabla del Diccionario de Datos que utiliza Oracle y los usuarios para realizar algunas operaciones. Contiene tan sólo una columna llamada **DUMMY** y una fila conteniendo el valor "X".



Diccionario de Datos

Vistas (sobre esas tablas):

- Pertenecen a **SYSTEM**.
- Pueden ser utilizadas por el resto de los usuarios de la BD.
- Existen sinónimos públicos para acceder fácilmente a ellas.



Diccionario de Datos

USER_

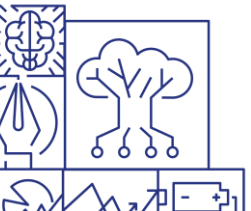
Objetos que pertenecen al propio usuario.

- No tienen la columna OWNER, ya que el propietario es el usuario actual.

ALL_

Todos los objetos accesibles por el usuario.

- Accesibles por el usuario por concesión explícita o pública de privilegios o roles.

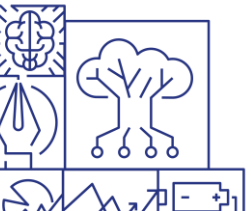


Diccionario de Datos

DBA_

Todos los objetos existentes (sólo para los DBA).

- Accesibles para los usuarios con el privilegio SELECT ANY TABLE.



Diccionario de Datos

USER_OBJECTS: Lista de todos los objetos pertenecientes al usuario (tablas, vistas, paquetes, índices, *triggers*, sinónimos, entre otros).

USER_TABLES: Lista de todas las tablas del usuario.

USER_VIEWS: Vistas del usuario.



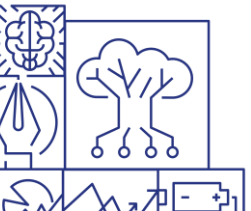
Diccionario de Datos

USER_USERS: Diversos datos sobre el usuario.

USER_UPDATABLE_COLUMNS: Columnas que pueden ser modificadas.

USER_JOBS: Tareas pertenecientes al usuario.

USER_TRIGGERS: Disparadores (triggers) del usuario.



Diccionario de Datos

USER_SYNONYMS: Sinónimos pertenecientes al usuario.

USER_INDEXES: Índices pertenecientes al usuario.

USER_CONSTRAINTS: Restricciones pertenecientes al usuario.

USER_CONS_COLUMNS: Columnas involucradas en cada restricción.



Diccionario de Datos

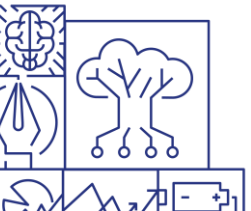
Tablas virtuales que almacenan información sobre la actividad de la BD durante su funcionamiento.

No son verdaderas tablas y no son accesibles para la mayoría de los usuarios. Sin embargo, los Administradores de la BD pueden consultar y crear **vistas** sobre esas tablas (llamadas *fixed views*) y autorizar el acceso a esas vistas por parte de otros usuarios.



Diccionario de Datos

El propietario de estas tablas es **SYS** y los nombres de las mismas comienzan por **V_\$**. Para estas tablas se crean unas vistas y para las vistas se crean sinónimos que comienzan todos por **V\$**.



Diccionario de Datos

- **V\$DATABASE** contiene información sobre la BD (nombre, entre otros).
- **V\$DATAFILE** contiene información sobre los *datafiles*.
- **V\$FIXED_TABLE** contiene información sobre todas las tablas de ejecución dinámica y vistas de la BD.



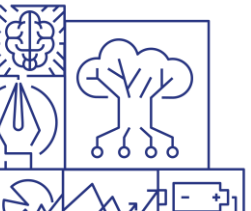
Diccionario de Datos

- **V\$SESSION** contiene información sobre las sesiones actuales (usuario comando actual, programa y terminal de acceso, entre otros).
- **V\$PROCESS** contiene información sobre los procesos activos.
- **V\$CONTROLFILE** contiene una lista con los ficheros de control.



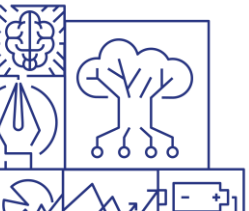
Oracle Database

- Estructuras lógicas:
 - Tablespace: la BD se divide en estructuras lógicas llamadas *tablespaces*, en la cual se agrupan estructuras relacionadas lógicamente. ORACLE crear por defecto varios tablespaces, pero requiere como mínimo el tablespace SYSTEM.

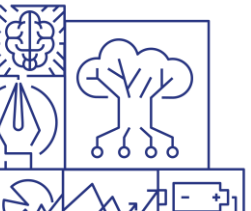
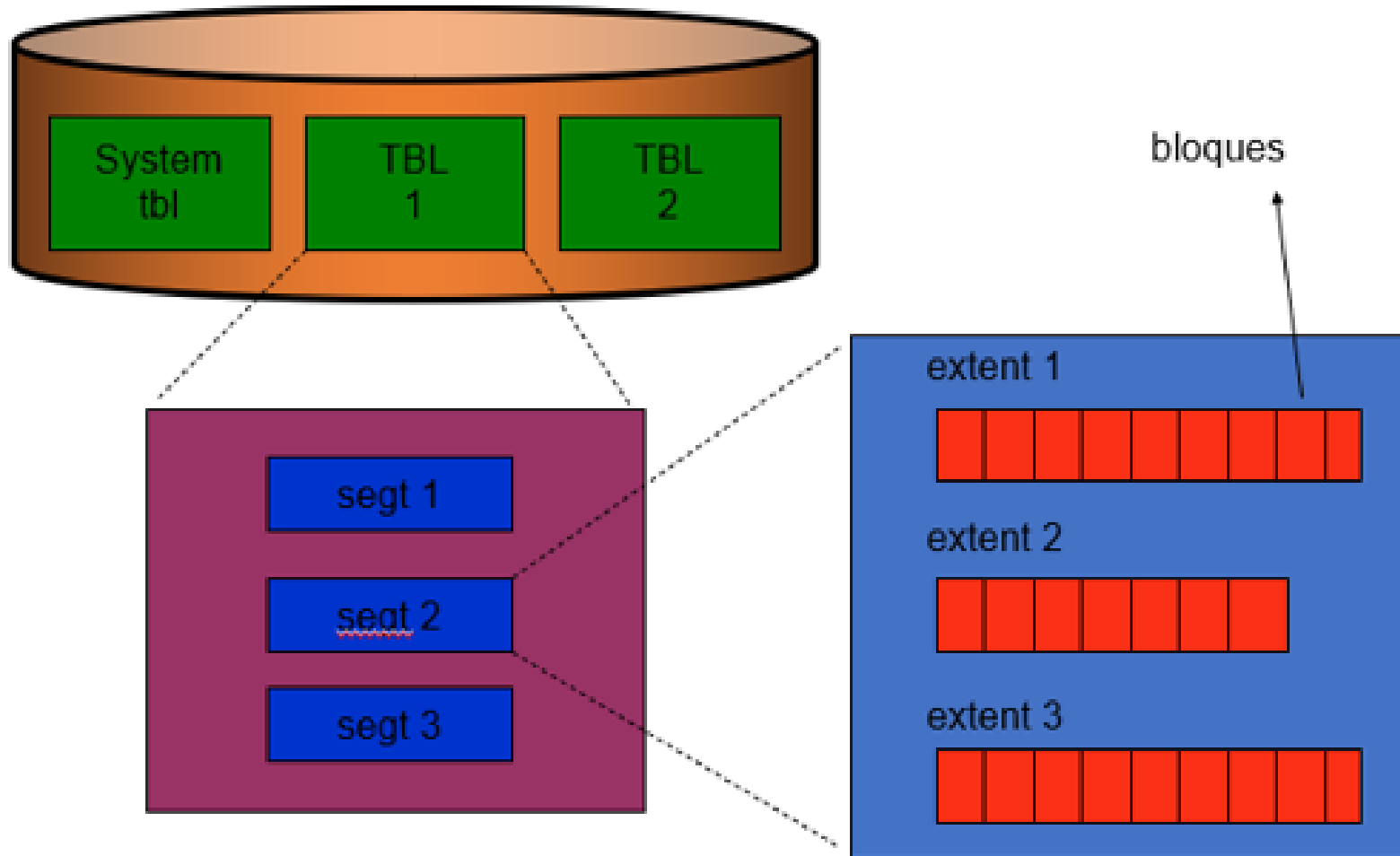


Oracle Database (Estructura)

- Segment: un segmento es un conjunto de extents asignados para estructuras lógicas como: tablas, índices, otros. Un segmento reside en un único tablespace y contiene como mínimo un extent.
- Extent: grupo de bloques continuos. Un segmento crece en extents.
- Block: unidad más pequeña de almacenamiento. Se define al crear la BD.



Oracle Database (Estructura)



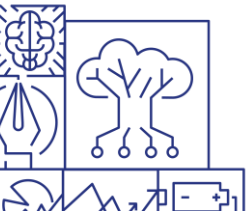
Oracle Database (Estructura)

- Estructuras Físicas:
 - Datafile: contienen los datos de la BD.
Corresponden a archivos físicos en el sistema operativo. Un datafile está asociado a un único tablespace, pero un tablespace puede contener mas de un datafile.



Oracle Database (Estructura)

- Control File: cada BD tiene al menos un controlfile, en él se almacena toda la información sobre la estructura física de la BD, nombres de cada datafile y redologs, al igual que el nombre de la base de datos, entre otra información.

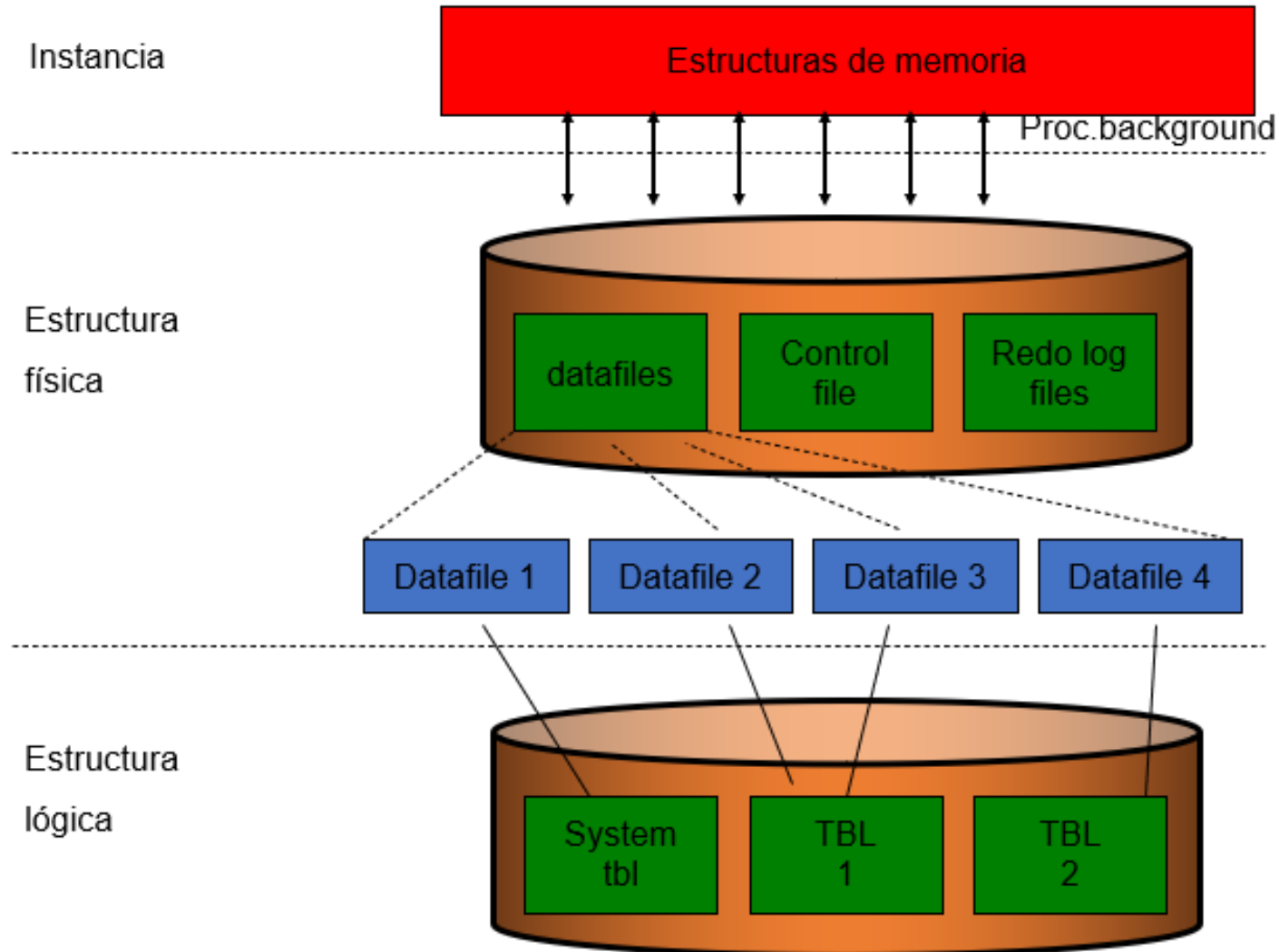


Oracle Database (Estructura)

- Redo logs: registran todos los cambios realizados a los datos. Cada BD debe tener dos o mas redo logs files, ya que ORACLE escribe en ellos en forma circular. Los redologs son críticos, ya que son utilizados para una recuperación después de una falla.



Oracle Database (Estructura)



Parámetros de Storage

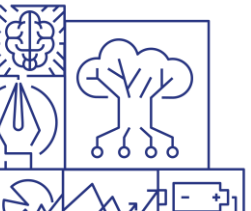
Corresponden a la forma en que crecerán nuestros segmentos y límites de los mismos.

- ✖ **INITIAL**: tamaño de nuestro extent inicial del segmento.
- ✖ **NEXT**: tamaño de los próximos extents en que crecerá el segmento.
- ✖ **PCTINCREASE**: usado para calcular porcentualmente, el tamaño de los subsiguientes extents.



Parámetros de Storage

- **MAXEXTENTS:** número máximo de extents que tendrá el segmento.
- **MINEXTENTS:** número mínimo de extents que tendrá el segmento.



Parámetros de Storage

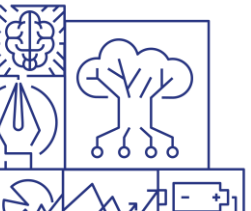
Si al crear un segmento, no se le definen los parámetros de storage, entonces toma los parámetros definidos por defecto para el tablespace.

- ✘ Create table tabla1 (campo1 number) storage (initial 1m next 1m pctincrease 0);
- ✘ Alter table tabla1 storage (next 20m)



Parámetros de Storage

- ✘ Create index indice1 on tabla1 (campo1) storage (initial 1m next 1m pctincrease 0);
- ✘ Alter index indice1 storage (next 20m)



NOTAS

- ✗ Todos los objetos en la BD tienen un OWNER.
- ✗ Si al crear el objeto no se le define el OWNER, entonces el propietario del nuevo objeto será el usuario con el que estoy conectado.
- ✗ Si al crear el objeto no se le define un tablespace, entonces el tablespace será el definido por default del usuario con que estoy conectado.

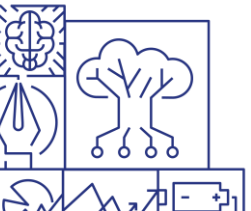


NOTAS

- ✘ Podemos invocar a tablas de diferentes propietarios al usuario que estamos conectados, si contamos con los permisos correspondientes.

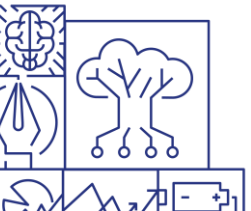
```
SELECT * FROM SCT.Tabla1;
```

- ✘ Siempre es recomendable indicar en programación el OWNER de las tablas.



NOTAS

- ✘ Create table tabla1 (campo1 number)
storage (initial 1m next 1m pctincrease 0);
- ✘ Create table SCT.tabla1 (campo1 number)
tablespace TBL_SCT
storage (initial 1m next 1m pctincrease 0);



NOTAS

✘ Create table tabla2 as SELECT * from SCT.tabla1;

Crea una tabla llamada tabla2 en el usuario actualmente conectado, es una copia exacta de la tabla1 del usuario SCT.



NOTAS

- ✘ Conviene por razones de eficiencia, separar las tablas de los índices en diferentes tablespaces, preferiblemente en unidades de discos independientes si fuera posible.
- ✘ No es recomendable crear objetos en los tablespaces propios de la BD como SYSTEM. Estos son reservados para objetos de diccionario de ORACLE.



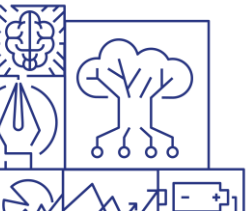
VISTAS IMPORTANTES EN ORACLE

- ✖ **dba_tablespace**: información sobre los tablespaces creados en la BD
- ✖ **dba_data_files**: información sobre los datafiles creados en la BD. En esta vista podemos ver el tamaño de nuestra BD.
- ✖ **dba_tables**: información sobre todas la tablas de nuestra BD.
- ✖ **user_tables**: información sobre las tablas del usuario con el que estamos conectados



VISTAS IMPORTANTES EN ORACLE

- ✖ **dba_segments**: información todos sobre los segmentos creados en la BD. El campo segment_type nos indicará el tipo de segmento.
- ✖ **dba_indexes**: información sobre todos los índices creados en la BD.
- ✖ **dba_views**: información sobre todas las vistas de la BD, importante para ver las vistas del diccionario de datos de ORACLE.



VISTAS IMPORTANTES EN ORACLE

- ✖ **v\$controlfile**: información de los archivos de control de la BD
- ✖ **v\$logfile**: información sobre los archivos log de la BD



VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_users (lista de usuarios de la BD)

```
-----
```

USERNAME	VARCHAR2(30)	Login del usuario
USER_ID	NUMBER	ID del usuario
DEFAULT_TABLESPACE	VARCHAR2(30)	tablespace por defecto
TEMPORARY_TABLESPACE	VARCHAR2(30)	tablespace temporal



VISTAS IMPORTANTES EN ORACLE

SQL> desc **dba_tablespaces** (lista de tablespaces de la BD)

```
-----
```

TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
INITIAL_EXTENT	NUMBER	valor en bytes del initial
NEXT_EXTENT	NUMBER	valor en bytes del next
MIN_EXTENTS	NUMBER	valor mínimo de extents
MAX_EXTENTS	NUMBER	valor máximo de extents
PCT_INCREASE	NUMBER	valor del pct_increase

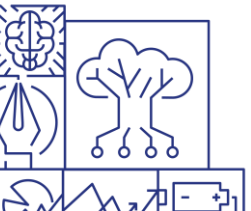


VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_data_files (lista de datafiles de la BD)

```
-----
```

FILE_NAME	VARCHAR2(513)	Nombre del datafile
FILE_ID	NUMBER	ID del datafile
TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
BYTES	NUMBER	bytes del archivo
BLOCKS	NUMBER	bloques del archivo



VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_tables (lista de todas las tablas de la BD)

OWNER	VARCHAR2(30)	Dueño de la tabla
TABLE_NAME	VARCHAR2(30)	Nombre de la tabla
TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
INITIAL_EXTENT	NUMBER	valor en bytes del initial
NEXT_EXTENT	NUMBER	valor en bytes del next
PCT_INCREASE	NUMBER	valor del pct_increase



VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_indexes (lista de todos los índices en la BD)

```
-----
```

OWNER	VARCHAR2(30)	Dueño del índice
INDEX_NAME	VARCHAR2(30)	Nombre del índice
TABLE_OWNER	ARCHAR2(30)	Dueño de la tabla indexada
TABLE_NAME	VARCHAR2(30)	Nombre de la tabla indexada
TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
INITIAL_EXTENT	NUMBER	valor en bytes del initial
NEXT_EXTENT	NUMBER	valor en bytes del next
PCT_INCREASE	NUMBER	valor del pct_increase



VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_free_space (Lista de extents libres de todos los tablespaces.
Pueden ser varios registros por tablespace)

```
-----
```

TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
FILE_ID	NUMBER	ID del datafile que contiene extent
BYTES	NUMBER	valor en bytes del extent
BLOCKS	NUMBER	cantidad de bloques



VISTAS IMPORTANTES EN ORACLE

SQL> desc dba_segments (lista de todos los segmentos de la BD)

OWNER	VARCHAR2(30)	Dueño del segmento
SEGMENT_NAME	VARCHAR2(81)	Nombre del segmento
SEGMENT_TYPE	VARCHAR2(18)	Tipo del segmento
TABLESPACE_NAME	VARCHAR2(30)	Nombre del tablespace
BYTES	NUMBER	Tamaño en bytes del segmento
BLOCKS	NUMBER	Tamaño en bloques del seg
EXTENTS	NUMBER	No. de extensiones del seg.
INITIAL_EXTENT	NUMBER	valor en bytes del initial
NEXT_EXTENT	NUMBER	valor en bytes del next_extent
MIN_EXTENTS	NUMBER	valor en bytes del min_extents
MAX_EXTENTS	NUMBER	valor en bytes del max_extents
PCT_INCREASE	NUMBER	valor del pct_increase

CREACION Tablespaces

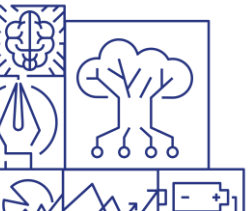
- ✘ create tablespace PRUEBA datafile
'c:\prueba1.dbf ' size 20M default storage (initial
1m next 1m pctincrease 0);
- ✘ alter tablespace PRUEBA add datafile
'c:\prueba2.dbf ' size 30M;
- ✘ alter database datafile 'c:\prueba2.dbf ' resize
50M;



Borrar tablespaces

- drop tablespace PRUEBA including contents <cascade constraints>;

Cuando se borra un tablespace, se deben borrar manualmente los datafiles a nivel del sistema operativo.



CREACION USUARIOS

✘ create user *usuario* identified by *clave* default tablespace T_SCT

temporary tablespace TEMP;

✘ Un usuario recién creado no posee permisos ni para conectarse, deben otorgarse ciertos privilegios.

✘ GRANT CONNECT TO usuario;

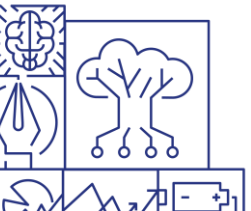
✘ GRANT RESOURCE TO usuario;

✘ ALTER USER usuario QUOTA UNLIMITED ON t_sct;

✘ GRANT DBA TO usuario;



PRACTICA





PROXIMA CLASE: CREACION DE UNA BASE DE DATOS

