



# SC-503

## ADMINISTRACIÓN DE BASE DE DATOS

Semana 1

## Definición

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

- Corresponde a las estructuras y los procesos de background usados para acceder los datos.

## Base de datos

- En ORACLE el término de base de datos se usa para representar los archivos físicos donde se almacena información.

## Estructura de la BD



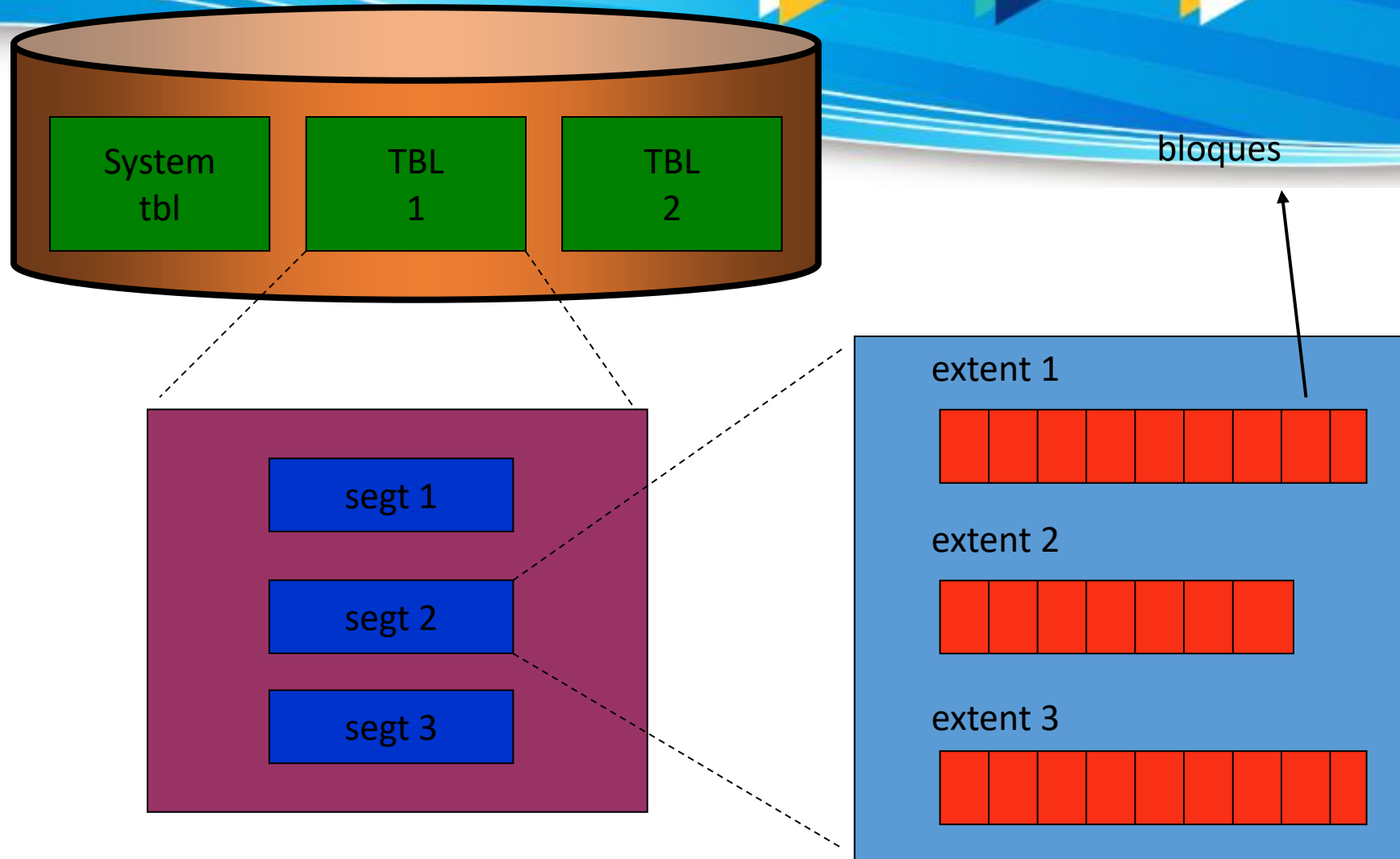
- Estructuras lógicas:
  - Tablespace: la BD se divide en estructuras lógicas llamadas *tablespaces*, en la cual se agrupan estructuras relacionadas lógicamente. ORACLE crear por defecto varios tablespaces, pero requiere como mínimo el tablespace SYSTEM.



## Estructura de la BD



- 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.



## Estructura de la BD



- 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.

## Estructura de la BD



- 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.



## Estructura de la BD



- 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.

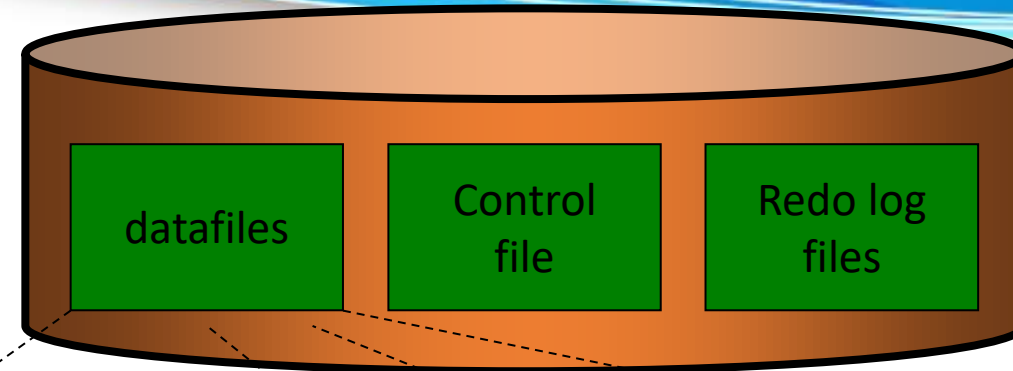
Instancia

Estructuras de memoria

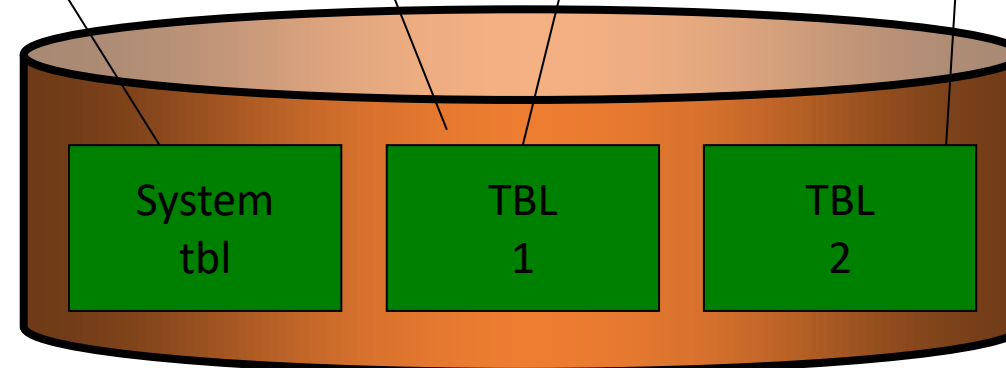
Proc.background



Estructura  
física



Estructura  
lógica



# 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.

- ✘ 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.





✘ 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);

✖ 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.



- ✘ 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;



# RESPALDO EN FRÍO



Respaldar toda la BD en frío, copiando los archivos del SO:

- ✖ connect sys as sysdba
- ✖ shutdown immediate
- ✖ copiar archivos:
  - ✖ dba\_data\_files, v\$logfile, v\$controlfile
- ✖ startup



# FIN