

MULTITENANT

MODULO III SESION 04

Ing. César Hjar

chijar@galaxy.edu.pe



AGENDA

GESTION DE CONSTRAINTS

- a) Integridad de datos.
- b) Tipos de constraints.
- c) Reglas y definición de restricciones.
- d) NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY y CHECK
- e) Habilitación y eliminación de restricciones.

INTRODUCCION AL RESPALDO DE INFORMACIÓN

- a) Importancia de la información en la base de datos.
- b) Configuración de base de datos para operaciones de backup y recovery.
- c) Uso inicial del recovery manager. (RMAN).
- d) Definición del Fast Recovery Area.
- e) Beneficios de usar el FRA.



CONSTRAINTS

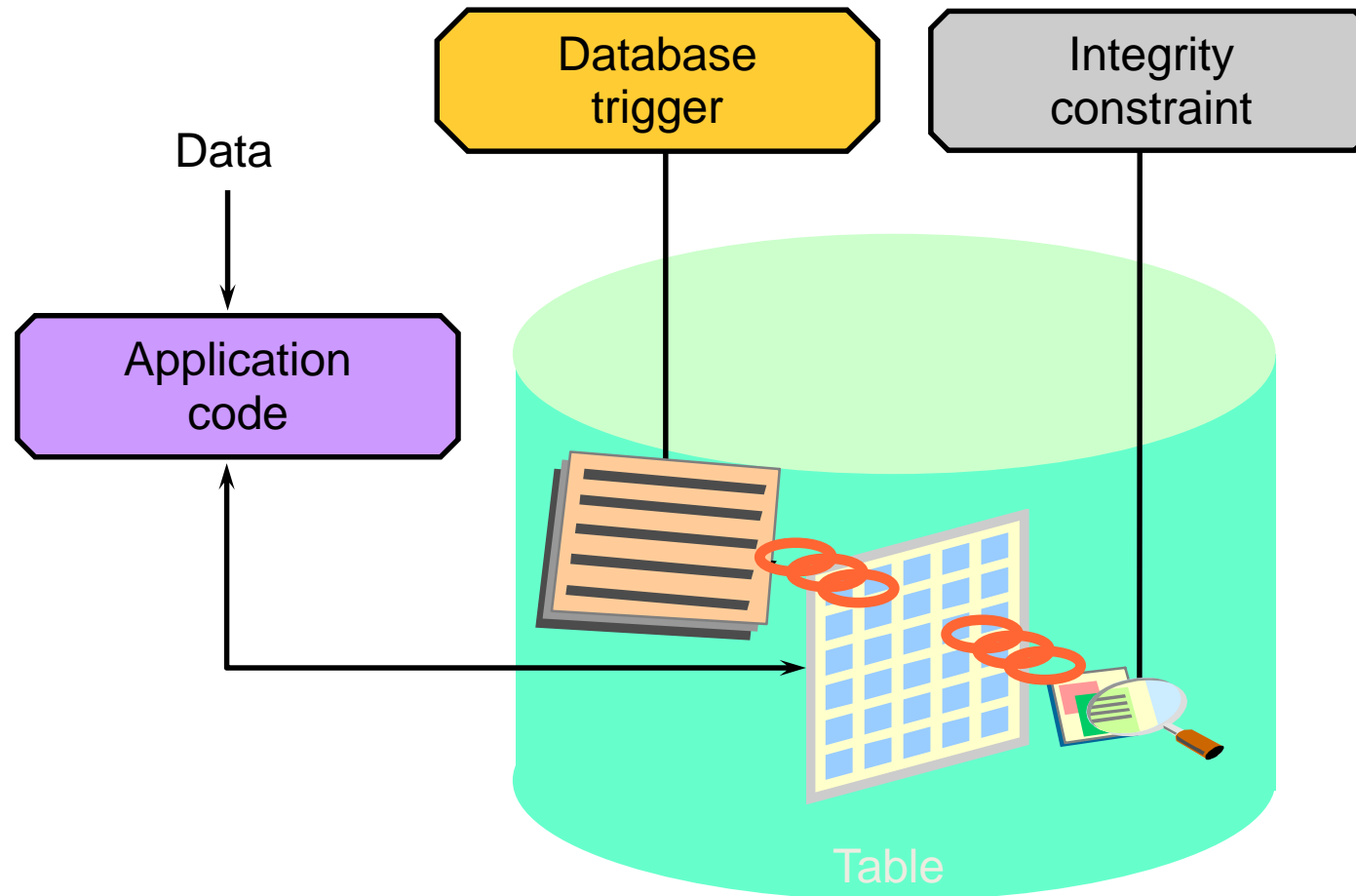


Objetivos de la Sesión

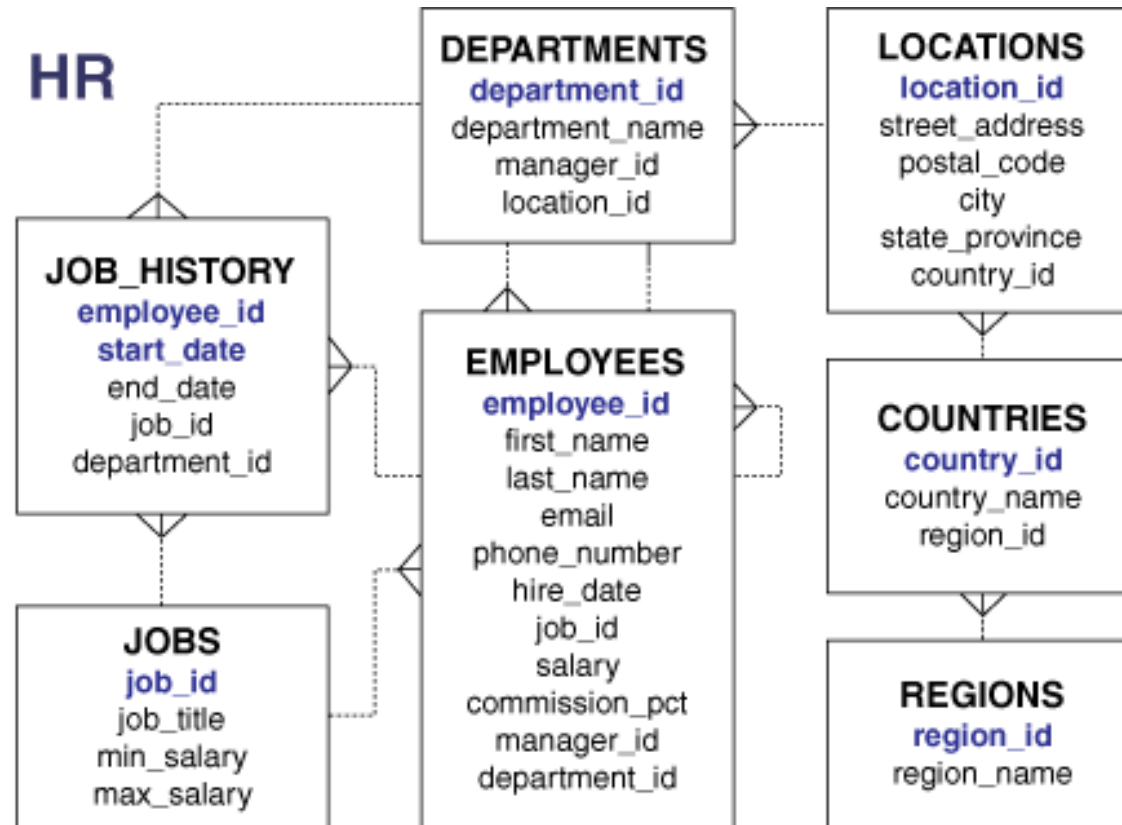
1. Definir los diferentes tipos de restricciones (constraints) que existen.
2. Consultar a las vistas del Diccionario de Datos para obtener la información de las restricciones.
3. Crear, modificar y eliminar restricciones.



Integridad de datos



Integridad de datos



Incluyendo restricciones

Las restricciones hacen cumplir reglas a nivel de tabla.

Las restricciones impiden la supresión de una tabla si hay dependencias.

Los siguientes tipos de restricciones son válidas:



NOT NULL

UNIQUE

PRIMARY KEY

FOREIGN KEY

CHECK

Tipos de restricciones

Restricción	Descripción
NOT NULL	Especifica que una columna no puede contener valores nulos
UNIQUE	Designa una columna o combinación de columnas como único
PRIMARY KEY	Designa una columna o combinación de columnas como clave primaria de la tabla
FOREIGN KEY	Designa una columna o combinación de columnas como clave externa, en una restricción de integridad referencial
CHECK	Especifica una condición que cada fila de la tabla debe satisfacer



Reglas para las restricciones

1. Se puede nombrar a una restricción, de no hacerlo el servidor de Oracle genera un nombre con el formato SYS_Cn.
2. Crear una restricción en cualquiera de los siguientes momentos:
 1. Al mismo tiempo que se crea la tabla
 2. Después que la tabla se ha creado
3. Definir una restricción a nivel de columna o tabla.
4. Ver una restricción en el diccionario de datos.



Definiendo restricciones

Sintaxis:

```
CREATE TABLE [schema.]table  
    (column datatype [DEFAULT expr]  
     [column_constraint],  
     ...  
     [table_constraint][,...]);
```

Restricción a nivel de columna:

```
column [CONSTRAINT constraint_name] constraint_type,
```

Restricción a nivel de tabla:

```
column, ...  
    [CONSTRAINT constraint_name] constraint_type  
    (column, ...),
```



Definiendo restricciones

Restricción a nivel de columna:

```
CREATE TABLE employees(  
    employee_id  NUMBER(6)  
        CONSTRAINT emp_emp_id_pk PRIMARY KEY,  
    first_name   VARCHAR2(20),  
    ...);
```

Restricción a nivel de tabla:

```
CREATE TABLE employees(  
    employee_id  NUMBER(6),  
    first_name   VARCHAR2(20),  
    ...  
    job_id       VARCHAR2(10) NOT NULL,  
    CONSTRAINT emp_emp_id_pk  
        PRIMARY KEY (EMPLOYEE_ID));
```



Restricción NOT NULL

Asegura que no se permiten valores nulos para la columna:

EMPLOYEE_ID	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	DEPARTMENT_ID
100	King	SKING	515.123.4567	17-JUN-87	AD_PRES	24000	90
101	Kochhar	NKOCHHAR	515.123.4568	21-SEP-89	AD_VP	17000	90
102	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	17000	90
103	Hunold	AHUNOLD	590.423.4567	03-JAN-90	IT_PROG	9000	60
104	Ernst	BERNST	590.423.4568	21-MAY-91	IT_PROG	6000	60
178	Grant	KGRANT	011.44.1644.429263	24-MAY-99	SA_REP	7000	
200	Whalen	JWHALEN	515.123.4444	17-SEP-87	AD_ASST	4400	10

...
20 rows selected.

↑
Restricción NOT NULL
(La fila no puede
contener valores
nulos para ésta
columna)

↑
**Restricción
NOT NULL**

↑
**Ausencia de
restricción NOT NULL**
(Cualquier fila puede
tener valores nulos
para ésta columna)



Restricción UNIQUE

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	EMAIL
100	King	SKING
101	Kochhar	NKOCHHAR
102	De Haan	LDEHAAN
103	Hunold	AHUNOLD
104	Ernst	BERNST

...

 INSERT INTO

208	Smith	JSMITH
209	Smith	JSMITH

← Allowed
← Not allowed:
already exists

UNIQUE constraint
↓

Restricción UNIQUE

Son definidas a nivel de columna o tabla

```
CREATE TABLE employees(  
    employee_id      NUMBER(6),  
    last_name        VARCHAR2(25) NOT NULL,  
    email            VARCHAR2(25),  
    salary            NUMBER(8,2),  
    commission_pct   NUMBER(2,2),  
    hire_date        DATE NOT NULL,  
    ...  
    CONSTRAINT emp_email_uk UNIQUE(email));
```



Restricción PRIMARY KEY

DEPARTMENTS

PRIMARY KEY

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500

...

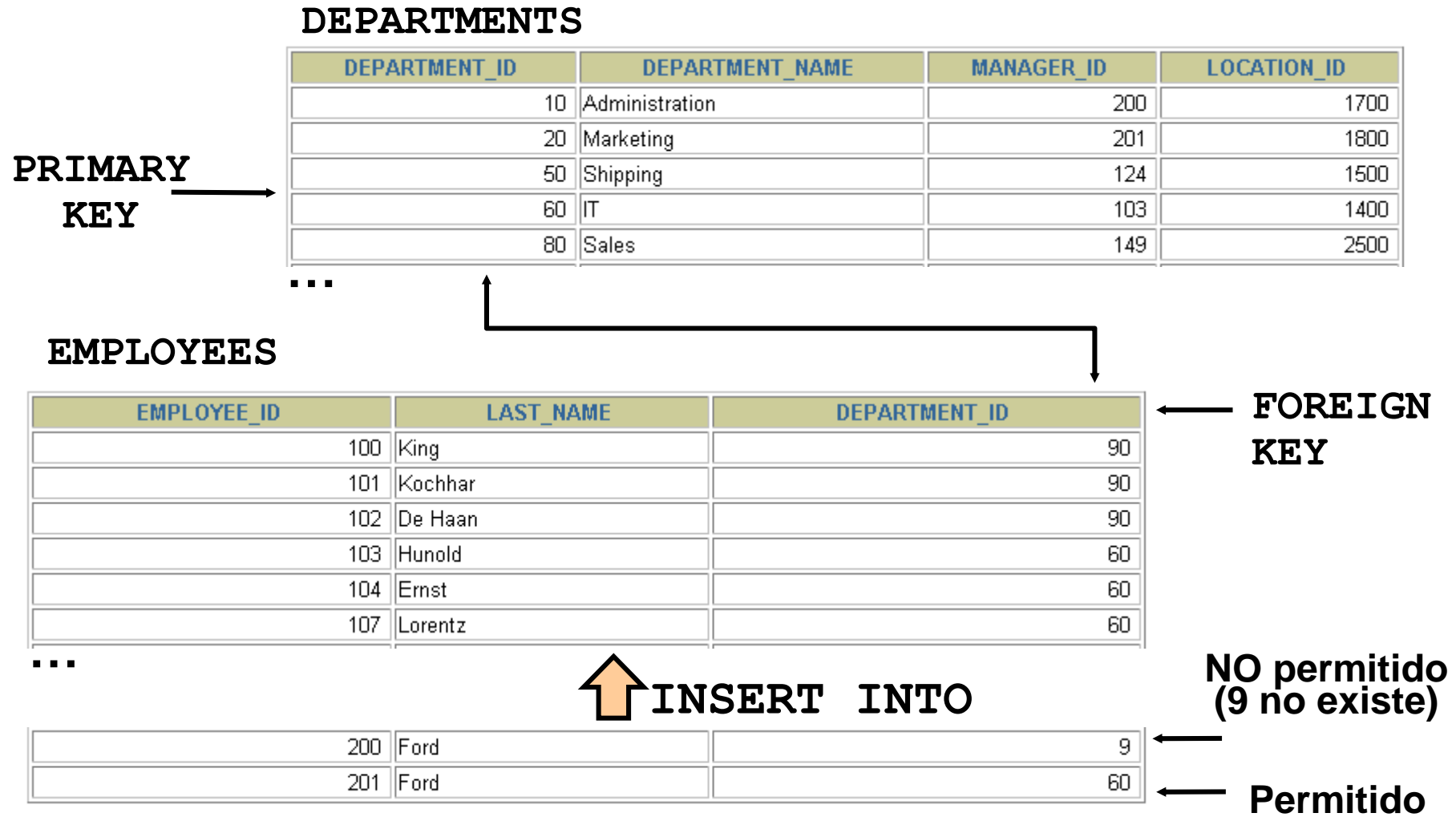
NO permitido
(null value)

↑ INSERT INTO

	Public Accounting		1400
50	Finance	124	1500

NO permitido
(50 ya existe)

Restricción FOREIGN KEY



Restricción FOREIGN KEY

Son definidas a nivel de columna o tabla

```
CREATE TABLE employees (  
    employee_id      NUMBER(6),  
    last_name        VARCHAR2(25) NOT NULL,  
    email            VARCHAR2(25),  
    salary            NUMBER(8,2),  
    commission_pct   NUMBER(2,2),  
    hire_date        DATE NOT NULL,  
    ...  
    department_id    NUMBER(4),  
    CONSTRAINT emp_dept_fk FOREIGN KEY (department_id)  
        REFERENCES departments(department_id),  
    CONSTRAINT emp_email_uk UNIQUE(email));
```



Restricción FOREIGN KEY

FOREIGN KEY: Define la columna en la tabla hija al nivel de restricción de tabla

REFERENCES: Identifica la tabla y la columna en la tabla padre

ON DELETE CASCADE: Elimina las filas dependientes en la tabla hija cuando una fila en la tabla padre es eliminada

ON DELETE SET NULL: convierte las dependencias de la clave foranea a valores nulos



Restricción CHECK

1. Define una condición que cada fila debe cumplir
2. Las siguientes expresiones no son permitidas:
 - a) Referencia a pseudo columnas **CURVAL**, **NEXTVAL**, **LEVEL** y **ROWNUM**
 - b) Llamar a funciones **SYSDATE**, **UID**, **USER** y **USERENV**
 - c) Consultas que hacen referencias a otros valores en otras tablas

```
..., salary  NUMBER(2)
      CONSTRAINT emp_salary_min
      CHECK (salary > 0), ...
```

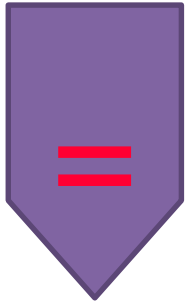


Estado de las restricciones

DISABLE
NOVALIDATE



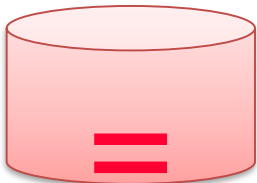
DISABLE
VALIDATE



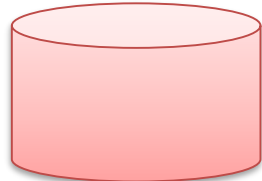
ENABLE
NOVALIDATE



ENABLE
VALIDATE



New data



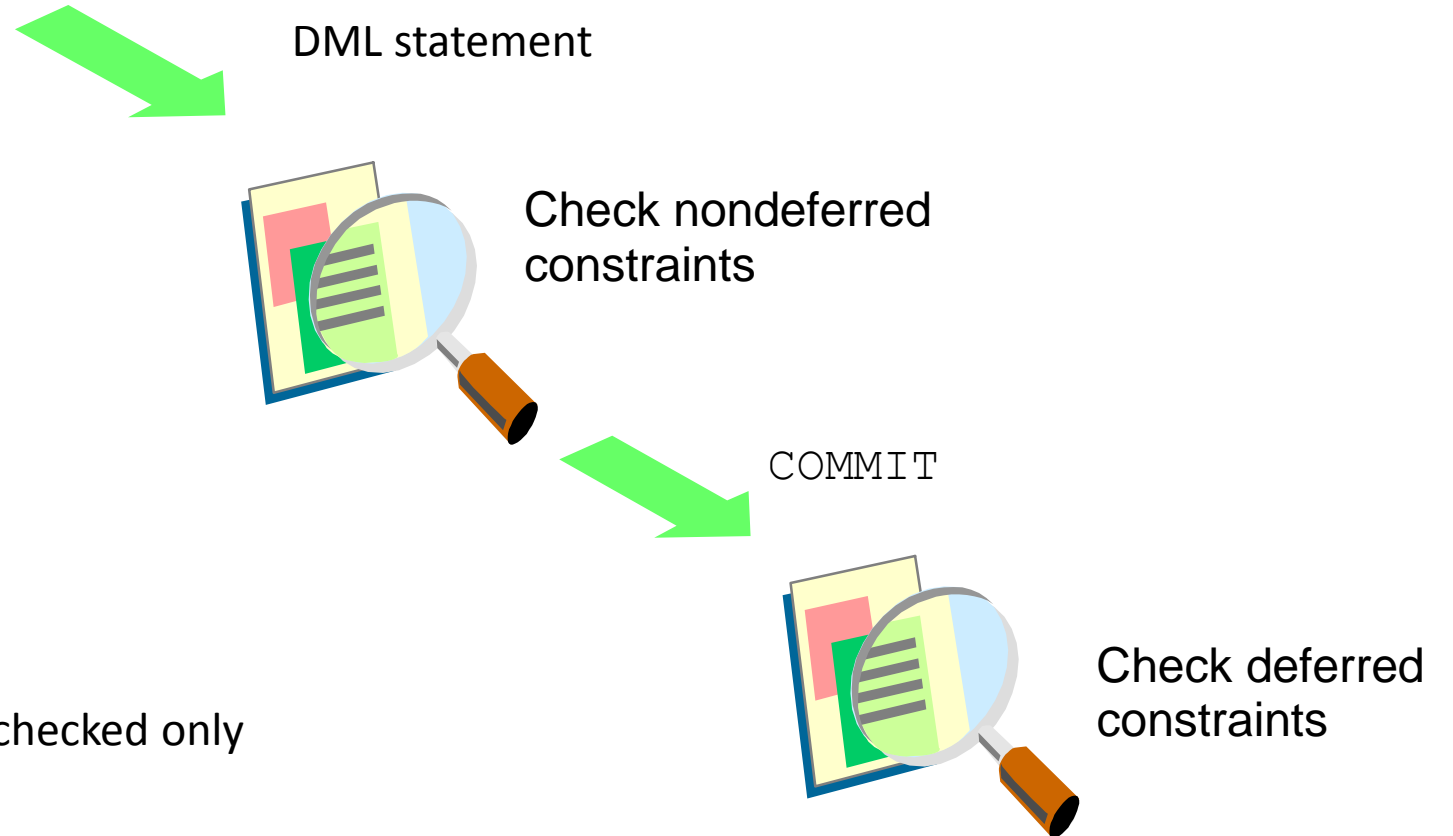
Existing data

- enabled-validate : comprueba los posteriores ingresos y actualizaciones y los valores existentes.
- disable-validate : no comprueba las posteriores inserciones y actualizaciones; pero sí comprueba los valores existentes.
- enabled-novalidate : si comprueba los posteriores ingresos y actualizaciones; pero no comprueba los datos existentes.
- disabled-novalidate : no comprueba los posteriores ingresos y actualizaciones, ni los valores existentes.

Comprobación de restricciones

Nondeferred or immediate constraints:

Nondeferred constraints, also known as immediate constraints, are enforced at the end of every DML statement.



Deferred constraints:

Deferred constraints are constraints that are checked only when a transaction is committed

Habilitando restricciones

ENABLE
NOVALIDATE

- No locks on table
- Primary and unique keys must use nonunique indexes

```
ALTER TABLE hr.departments  
ENABLE NOVALIDATE CONSTRAINT dept_pk;
```

Habilitando restricciones



ENABLE
VALIDATE

- Locks the table
- Can use unique or nonunique indexes
- Needs valid table data

```
ALTER TABLE hr.employees  
ENABLE VALIDATE CONSTRAINT emp_dept_fk;
```

Eliminando una restricción

Para eliminar una restricción se utiliza la siguiente sintaxis:

```
ALTER TABLE hr.employees  
drop CONSTRAINT emp_dept_fk;
```



Obteniendo información de las restricciones

Para obtener información acerca de las restricciones se pueden consultar las siguientes vistas:

DBA_CONSTRAINTS

DBA_CONS_COLUMNS



Resumen

1. Existen 5 tipos de constraints los cuales deben ser usados según la validación que se desea aplicar.
2. Se recomienda que las tablas tengan una Llave Primaria para evitar la duplicidad de los Datos.
3. Los PK y FK unidos permiten contar con datos fiables lo cual se conoce como Integridad de los Datos.
4. Existen 2 formas de crear restricciones: a nivel de columna y a nivel de tabla.



Información adicional

How to use Primary key, Foreign Key, Check, Not Null, Unique Integrity constraints in Oracle:

http://www.oracle-dba-online.com/sql/integrity_constraints.htm

Finding Foreign Key Constraints in Oracle :

http://www.dbasupport.com/oracle/ora10g/foreign_key_constraints.shtml



RESPALDO Y RECUPERACION





GOBIERNO

Síguenos en Google News



OSCE: 5 claves para entender pérdida de casi 800 mil archivos

Información digital sobre contrataciones estatales almacenada en servidores de institución se perdió debido a negligencia



Objetivos de la Sesión

1. Definir las principales estrategias de respaldo de datos.
2. Analizar los diferentes escenarios de pérdida de información.
3. Implementar soluciones de backup de base de datos.



Pérdida de Información

- Lo único constante... es el cambio.
- Todo cambio contiene un riesgo inherente. Cualquier cosa puede suceder. Generalmente las cosas suceden como no se desea que ocurra.
- Debemos estar preparados para cualquier eventualidad.
- El riesgo siempre va a existir, es por esto que lo que se busca es minimizar lo, de tal forma que si ocurre, cause el menor daño posible.



Pérdida de Información

- La información es uno de los recursos mas preciados para las empresas.
- Debemos estar preparados para una eventual perdida de información.
- Para evitar que la información se pierda completamente por alguna razón, debemos contar con mecanismos apropiados que permitan restaurar la información, minimizando el impacto de la perdida.
- Oracle Server provee diferentes formas y mecanismos para realizar respaldos de información.



Categorías de Falla

Se han definido diferentes categorías para clasificar las posibles fallas, las cuales se muestran a continuación:

1. Falla de una sentencia
2. Falla de un proceso usuario
3. Error de Usuario
4. Falla de la red de datos
5. Falla de la instancia
6. Falla de hardware de almacenamiento



Categorías de Falla

Cada categoría de falla puede requerir diferentes tipos de estrategias de respaldos de información.

Por ejemplo:

Si existe una falla al momento de ejecutar una sentencia, esta puede ser muy fácilmente superada, inclusive sin la intervención del DBA.

Si se da algún tipo de falla como una perdida total del servidor, se debería utilizar una estrategia de recuperación de información.



Causas de Falla de una sentencia

- Error lógico en una aplicación.
- Intento de ingreso invalido de datos en una tabla.
- Intento de ejecutar una operación sin los suficientes privilegios.
- Intento de crear una tabla o de asignar un extent cuando ya no se tiene mas cuota en el tablespace.
- Intento de crear una tabla o de asignar un extent cuando ya no se tiene mas espacio en el tablespace.



Soluciones de Falla de una sentencia

- Corregir el flujo de instrucciones en una aplicación.
- Modificar y corregir sentencias SQL no adecuadas.
- Proveer los privilegios suficientes a los usuarios para que puedan realizar las acciones necesarias.
- Cambiar el limite de una cuota al usuario para que puedan alocar mas espacio en un tablespace.
- Agregar mas espacio o añadir un nuevo datafile a un tablespace.



Causas de falla de procesos de usuario

- Cuando el proceso usuario realiza una finalización anormal.
- Cuando la sesión del usuario termina de forma anómala.
- Cuando el proceso usuario genera un error debido a una excepción no manejada.



Soluciones de falla de procesos de usuario

- El Process Monitor (PMON) detecta la falla del proceso usuario.
- El Process Monitor (PMON) se encarga de hacer los rollbacks correspondientes y de liberar todos los recursos y bloqueos del usuario.



Posibles errores de usuario

```
SQL> DROP TABLE employees;
```

```
SQL> TRUNCATE TABLE employees;
```

```
SQL> DELETE FROM employees;  
SQL> COMMIT;
```

```
SQL> UPDATE employees  
2> SET salary = salary * 1.5;  
SQL> COMMIT;
```

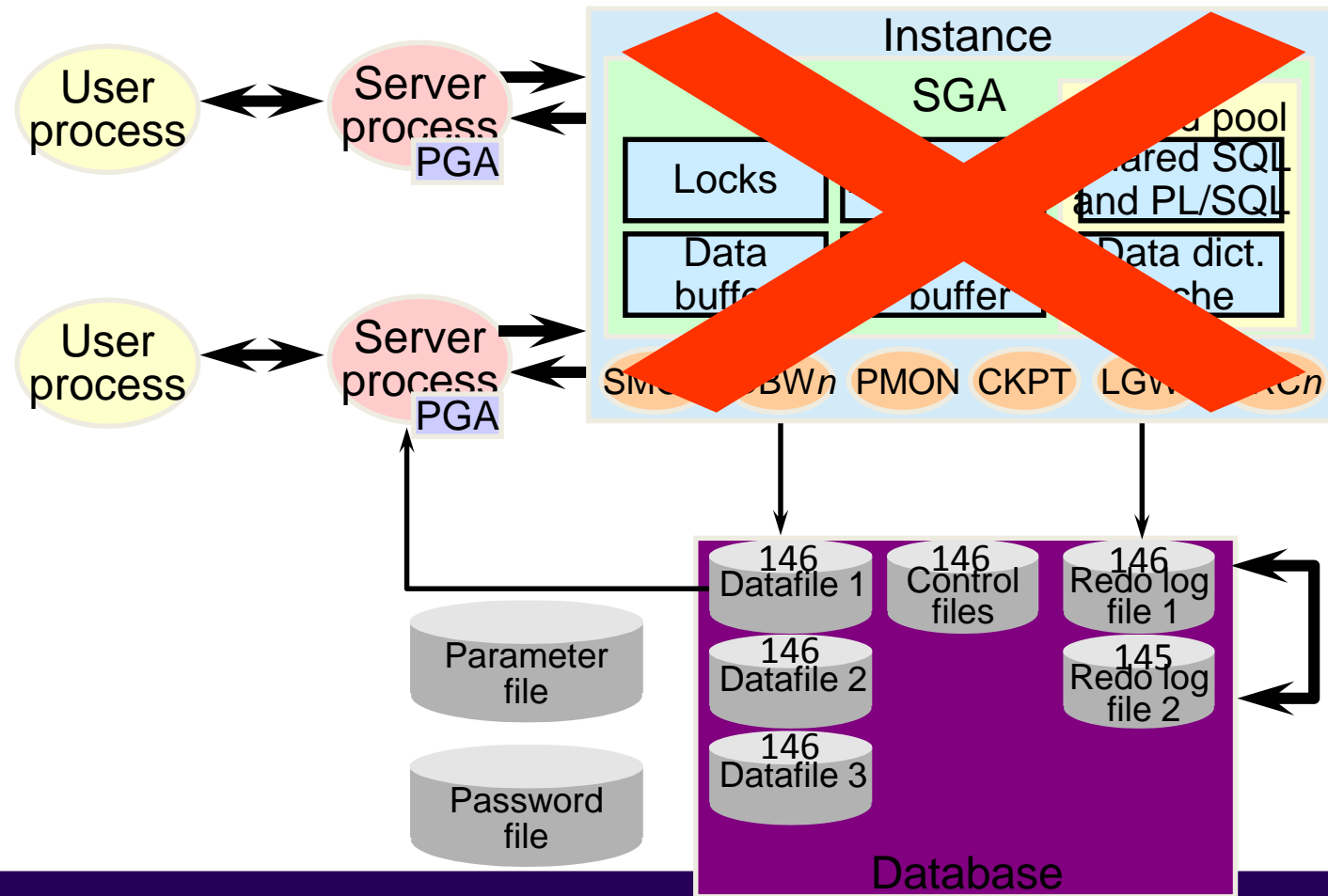


Solución de errores de usuario

- Entrenamiento a los usuarios, para disminuir el riesgo.
- Hacer una recuperación de la información desde un respaldo de información válido.
- Hacer un import de la tabla desde un export válido.
- Usar un Flashback query para recuperar la información.
- Usar el LogMiner para recuperar la información.



Causas de falla de una Instancia



Causas de falla de una Instancia de Base de Datos

- Falla eléctrica del servidor donde se encuentra levantada la instancia de base de datos.
- La instancia falla debido a problemas de hardware tales como CPU, memoria, etc.
- La instancia falla debido a que se ha caído el sistema operativo.
- Alguno de los background process (DBWn, LGWR, PMON, SMON, CKPT) experimenta una falla y se cae la instancia de base de datos.



Recuperación de una falla de Instancia

- No se requiere acciones especiales del DBA para recuperar información en caso una falla de la instancia de la base de datos.
- Iniciar la instancia y el System Monitor (SMON) automáticamente ejecutara las acciones necesarias para recuperar la base de datos.
- Esperar a que el mensaje “Database Open” se muestre en pantalla. Esto indica que la base de datos esta nuevamente en un estado consistente.
- Comunicar a los usuarios que todas las transacciones que fueron comprometidas, deben ser nuevamente ingresadas.



Causas de falla de hardware de almacenamiento

- Falla de los cabezales del disco duro.
- Problemas físicos en los platos del disco duro.
- Sobrecalentamiento del disco duro.
- El archivo es accidentalmente eliminado.



Soluciones de falla de hardware de almacenamiento

- Esto va depender en gran medida de la estrategia de respaldos de información que se ha definido para la base de datos.
- Lo mas común es restaurar desde el ultimo backup y aplicar los redo logs necesarios para recuperar toda la información.



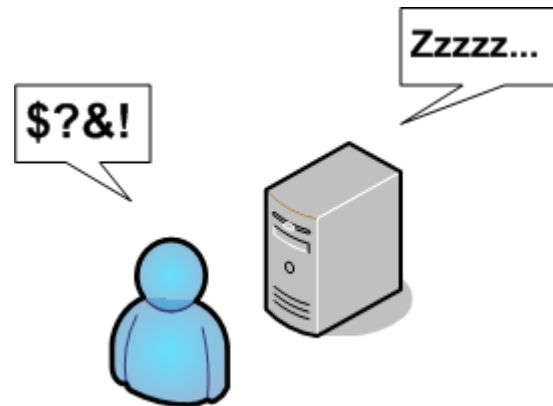
Definiendo una estrategia de respaldo

- Requerimientos del negocio
- Requerimientos operacionales
- Consideraciones técnicas
- Gestión de concurrencia

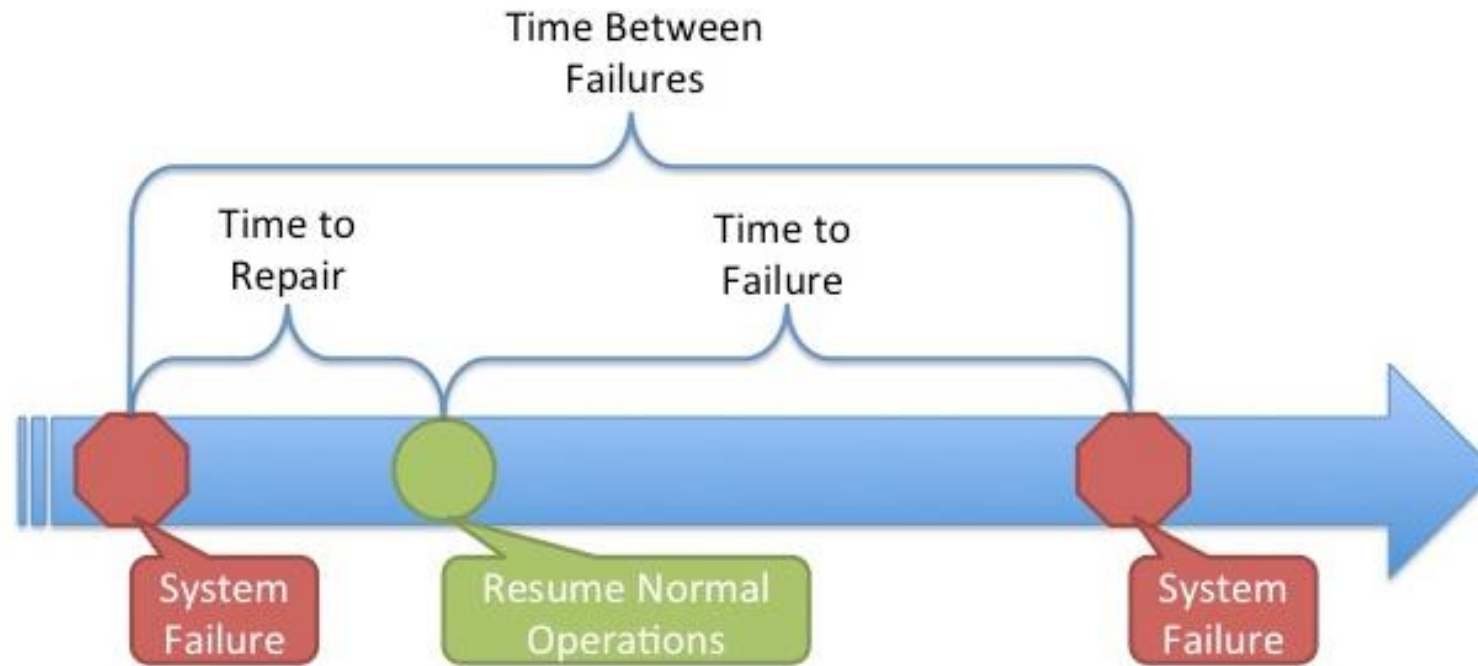


Requerimientos del negocio

- Tiempo medio para recuperar (MTTR)
- Tiempo medio entre fallos (MTBF)
- Proceso evolutivo



Requerimientos del negocio



Requerimientos operacionales

- Operación durante las 24 horas
- Pruebas y validación de las copias de seguridad
- Volatilidad de la base de datos



Consideraciones técnicas

- Recursos: hardware, software, mano de obra y tiempo
- Copias de imágenes físicas de los archivos del sistema operativo
- Copias lógica de los objetos en la base de datos
- Configuración de base de datos
- El volumen de transacciones que afecta a la frecuencia deseada de las copias de seguridad



Definiendo una estrategia de respaldo

- Existen diferentes tipos de estrategias de respaldos de información, tales como:
- Respalos en Frio: Se realizan cuando la base de datos se encuentra sin ser utilizada, es decir cuando la base de datos esta cerrada.
- Respalos en Caliente: Se realizan cuando la base de datos se encuentra en total operación.



RespalDOS manuales

Se debe tener en cuenta que los respaldos de información deben hacer backups de los siguientes componentes:

- Datafiles
- Controls Files
- Redo Logs Files
- Parameters Files
- Passwords Files



Obtener información de la base de datos



V\$DATAFILE

V\$CONTROLFILE

V\$LOGFILE

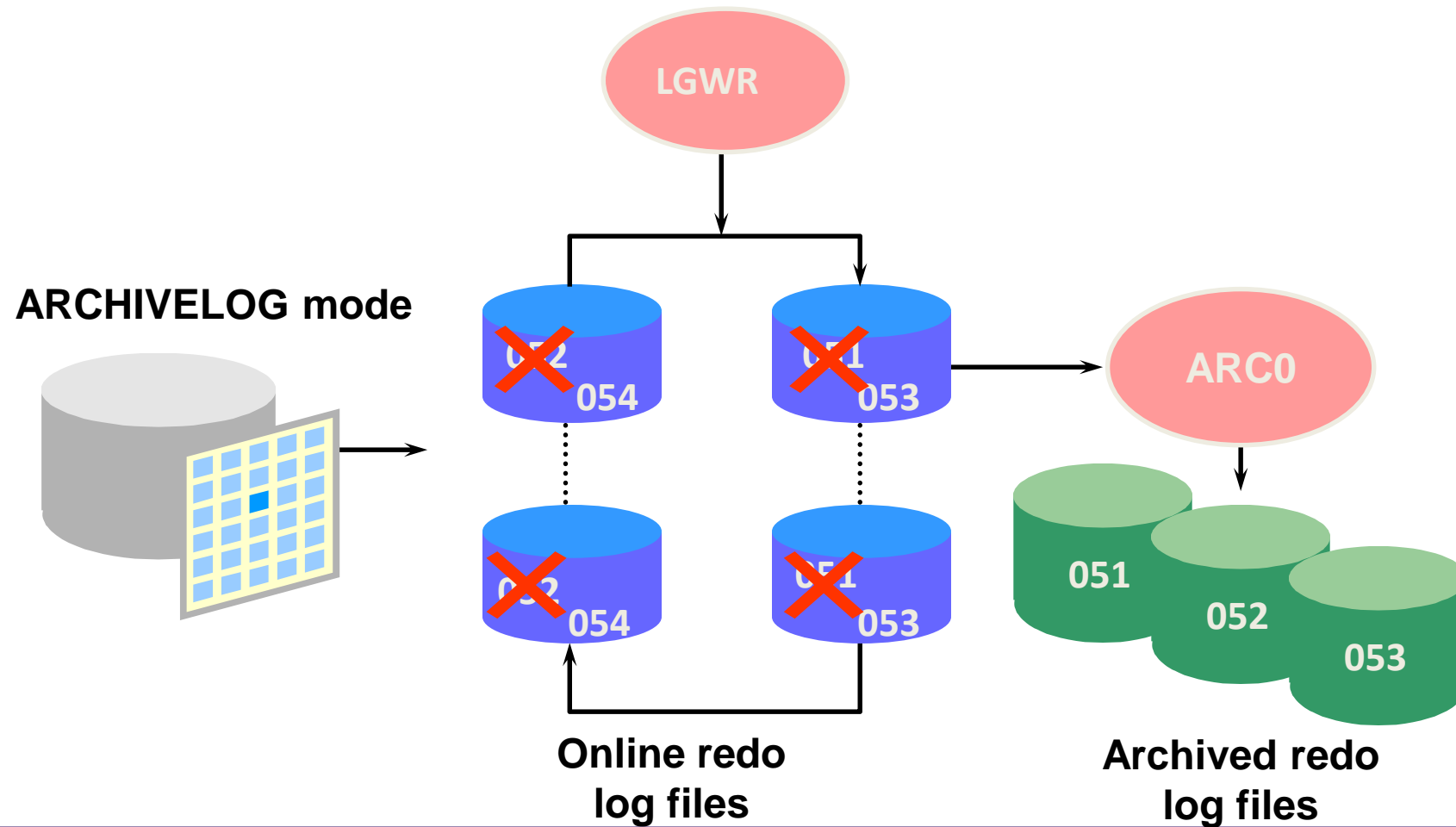
DBA_DATA_FILES

Ventajas de backup de base de datos abierta

- Mantiene una alta disponibilidad de la base de datos
- Se puede hacer a nivel del tablespace o el nivel de datafile
- Soporta operaciones de negocios sin parar

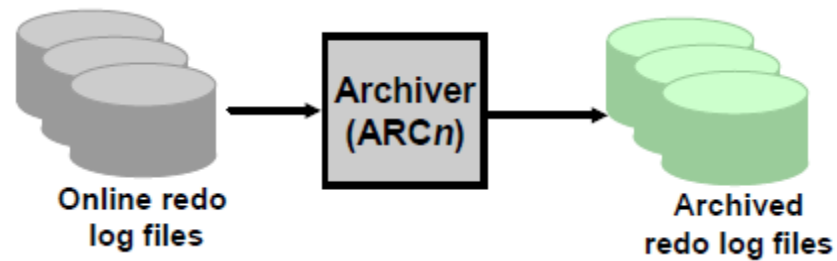


Requisitos para un backup de base de datos abierta

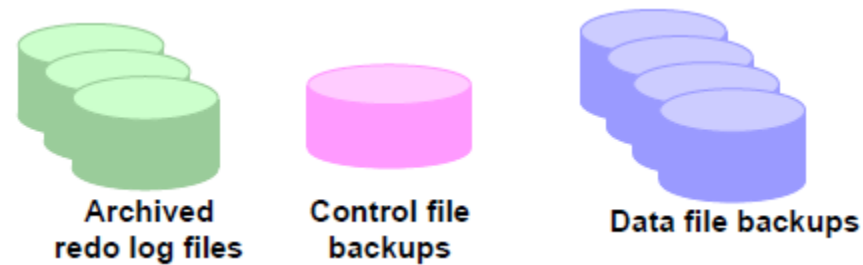


Configuración de la base de datos para operaciones de Backup y Recovery

Operar la base de datos en modo ARCHIVELOG

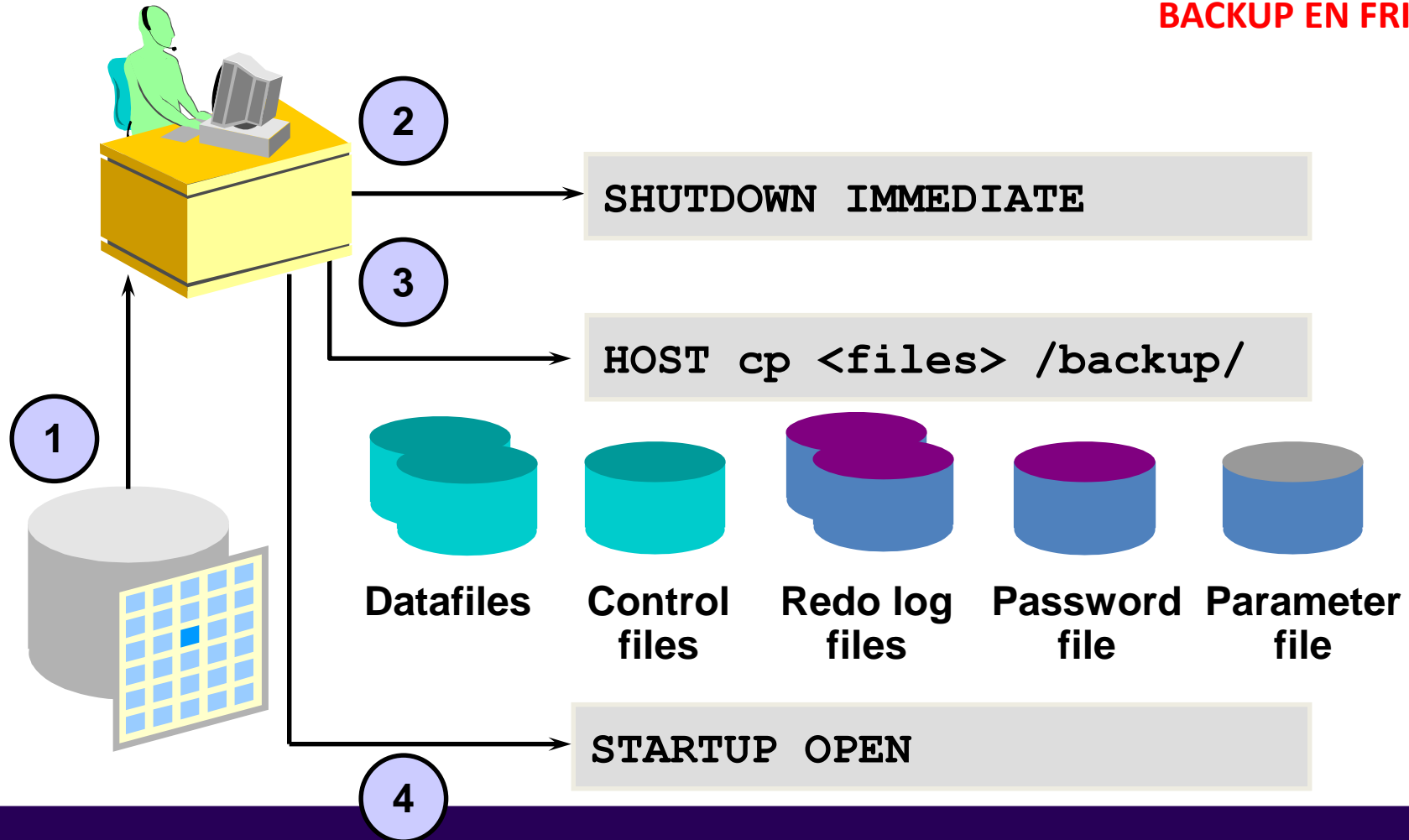


Configurar el FRA



Realizando un backup completo consistente

**COLD BACKUP –
BACKUP EN FRIO**



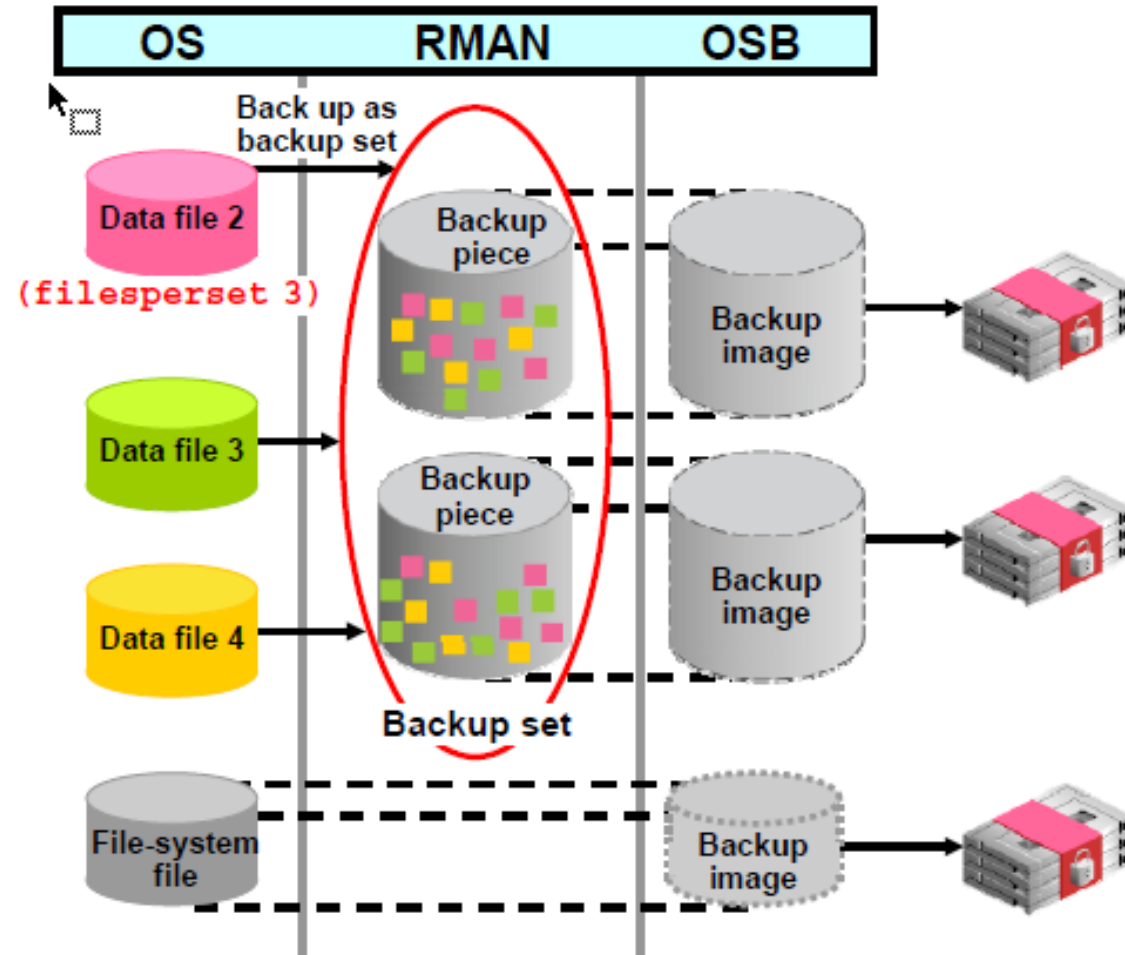
Configurando el modo ARCHIVELOG

Para colocar la base de datos en modo ARCHIVELOG, realice los siguientes pasos:

1. El uso de Enterprise Manager
 - a) Seleccione la casilla de verificación "Modo ARCHIVELOG".
 - b) Haga clic en Aplicar. La base de datos se puede establecer en el modo ARCHIVELOG sólo desde el estado MOUNT.
 - c) Haga clic en Sí cuando se le pregunte si desea reiniciar la base de datos.
2. El uso de comandos SQL
 - a) Monte la base de datos.
 - b) Emita el comando ALTER DATABASE ARCHIVELOG.
 - c) Abra la base de datos.



Soluciones de Backup Oracle



Utilizando el recovery manager

```
$ rman target /  
  
RMAN> BACKUP DATABASE;  
Starting backup at 10-JUN-07  
.  
.  
RMAN> LIST BACKUP;  
BS Key Type LV Size Device Type Elapsed Time Completion Time  
-----  
1 Full 1.06G DISK 00:01:49 10-JUN-07  
.  
.  
RMAN> DELETE OBSOLETE;  
.  
.  
Do you really want to delete the above objects (enter YES or NO)? YES  
deleted archived log  
.  
.
```



Especificación de formato

Sintaxis	Descripción
%d	Nombre de la base de datos
%T	Especifica el año, mes y día en el calendario Gregoriano en el format YYYYMMDD.
%t	Backupet timestamp
%s	Número de backupset
%p	Pieza de backup
%F	Combina el DBID, día, mes, año y secuencia en un único y repetido nombre autogenerado (c- IIIIIIIIII-YYYYMMDD-QQ)



Tipos de comandos RMAN

Los comandos RMAN son de los siguientes tipos:

Comando Stand-alone:

- Es ejecutado individualmente desde el prompt RMAN

- No puede aparecer como subcomandos dentro de RUN

Comando Job:

- Debe estar dentro de las llaves de un comando RUN

- Se ejecuta como un grupo

- Algunos comandos pueden ser ejecutados, ya sea como un comando independiente o de trabajo



Ejemplo de comando Job

```
RMAN> RUN
2> {
3>   ALLOCATE CHANNEL c1 DEVICE TYPE DISK
4>   FORMAT "/disk2/%U";
5>   BACKUP AS BACKUPSET DATABASE;
6>   SQL 'alter system archive log current';
7> }
```

La ejecución de todo el bloque se inicia cuando se introduce esta línea.

Desasignado después de que finalice el bloque de RUN



Usando el Fast Recovery Area

Elementos permanentes:

- Copias multiplexadas del archivo de control actual

- Copias multiplexadas de registros de rehacer en línea

(Online redo log)

Elementos transitorios:

- Registros de rehacer archivados

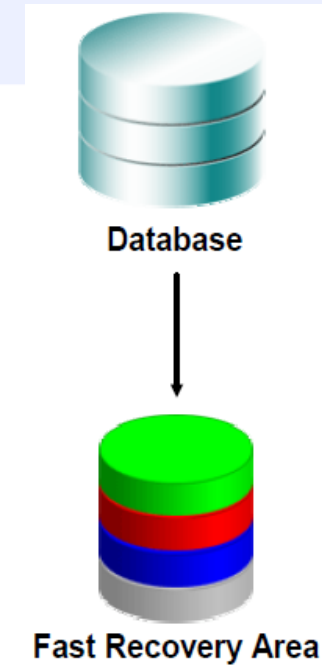
- Copias de archivos de datos

- Copias de archivos de control

- Autobackups de archivo de control

- Piezas de copia de seguridad

- Registros de Flashback



Provides a centralized storage location for backup and recovery files. Oracle creates archived logs and flashback logs in the fast recovery area. Oracle Recovery Manager (RMAN) can store its backup sets and image copies in the fast recovery area, and it uses it when restoring files during media recovery. The fast recovery area also acts as a disk cache for tape.

Definiendo el Fast Recovery Area

El FRA está definido seteando dos parámetros de inicialización:

DB_RECOVERY_FILE_DEST_SIZE: establece el límite de disco

DB_RECOVERY_FILE_DEST: establece la ubicación del FRA


☒ ARCHIVELOG Mode*

Log Archive Filename Format*

Number	Archived Redo Log Destination	Status	Type
1	<input type="text" value="USE_DB_RECOVERY_FILE_DEST"/>	VALID	Local

☒ **TIP** It is recommended that archived redo log files be written to multiple locations spread across the different disks.

☒ **TIP** You can specify up to 10 archived redo log destinations.

Flash Recovery Area Location 

Flash Recovery Area Size

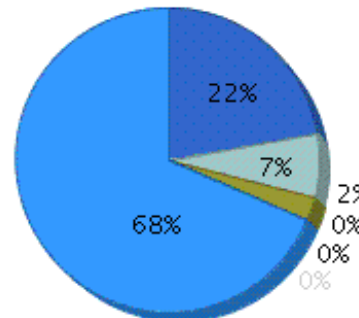
Flash Recovery Area Size must be set when the location is set.

Non-reclaimable Flash Recovery Area (GB) **1.89**

Reclaimable Flash Recovery Area (MB) **609**

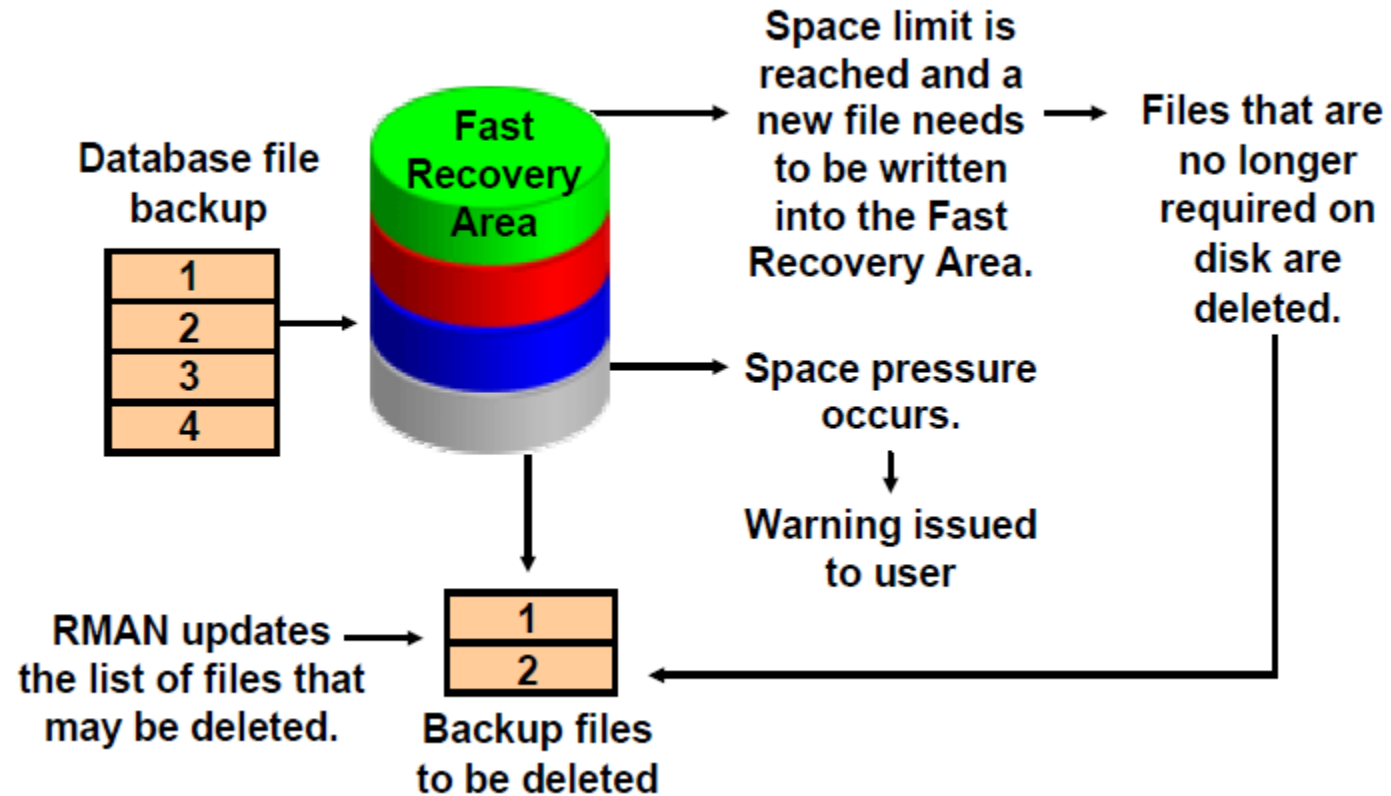
Free Flash Recovery Area (GB) **3.51**

Flash Recovery Area Usage



Usage Category	Percentage
Used (Total)	68%
Non-reclaimable	22%
Reclaimable	7%
Free	2%
Other	0%
Other	0%
Other	0%

Administración de espacio del FRA



Uso de espacio del FRA

- Configurar la directiva de retención con el valor adecuado mínimo para su base de datos.
- Copia de seguridad de los archivos de registro regularmente y eliminar los archivos una vez finalizada la copia de seguridad.
- Opcionalmente, configurar una política de eliminación de registro de rehacer de archivado.
- Utilice el REPORT OBSOLETE y DELETE OBSOLETE para eliminar las copias de seguridad y copias de archivos que no son necesarios.



Beneficios del uso del FRA

El uso del FRA para la recuperación de archivos relacionados:

- Simplifica la ubicación de los archivos de backup

- Administra automáticamente el espacio en disco destinado para los archivos de recuperación



Información adicional

Database Backup and Recovery User's Guide

[https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRA
DV8001](https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRA
DV8001)

Oracle Flashback Technology

[https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRA
DV89342](https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRA
DV89342)





GALAXY
TRAINING