

## 6. Manipulación de datos. INSERT, UPDATE y DELETE

### 6.5 ROLLBACK, COMMIT y AUTOCOMMIT



## 6.5 ROLLBACK, COMMIT y AUTOCOMMIT

Supongamos que queremos borrar una fila de una tabla pero, al teclear la orden SQL, se nos olvida la cláusula WHERE y... ¡horror!, ¡borramos todas las filas de la tabla! Esto no es problema, pues Oracle permite dar marcha atrás a un trabajo realizado mediante la **orden ROLLBACK**, siempre y cuando no hayamos validado los cambios en la base de datos mediante la **orden COMMIT**.

*\*Operaciones INSERT, UPDATE, DELETE*

Cuando hacemos **transacciones** sobre la base de datos, es decir, cuando insertamos, actualizamos y eliminamos datos en las tablas, los cambios no se aplicarán a la base de datos hasta que no hagamos un COMMIT. Esto significa que, si durante el tiempo que hemos estado realizando **transacciones**, no hemos hecho ningún COMMIT y de pronto se va la luz, todo el trabajo se habrá perdido, y nuestras tablas estarán en la situación de partida.

Para validar los cambios que se hagan en la base de datos tenemos que ejecutar la orden COMMIT:

```
SQL> COMMIT;  
Validación terminada.
```

SQL\*Plus e iSQL\*Plus permiten validar automáticamente las **transacciones** sin tener que indicarlo de forma explícita. Para eso sirve el **parámetro AUTOCOMMIT**. El valor de este parámetro se puede mostrar con la orden SHOW, de la siguiente manera:

```
SQL> SHOW AUTOCOMMIT;  
autocommit OFF
```

**OFF** es el valor por omisión, de manera que las **transacciones** (INSERT, UPDATE y DELETE) no son definitivas hasta que no hagamos COMMIT. Si queremos que INSERT, UPDATE Y DELETE tengan un carácter definitivo sin necesidad de realizar la validación COMMIT, hemos de activar el parámetro AUTOCOMMIT con la **orden SET**:

```
SQL> SET AUTOCOMMIT ON;
```

```
SQL> SHOW AUTOCOMMIT;  
autocommit IMMEDIATE
```

Ahora, cualquier INSERT, UPDATE y DELETE se validará automáticamente.

La orden **ROLLBACK** aborta la transacción volviendo a la situación de las tablas de la base de datos desde el último COMMIT:

```
SQL> ROLLBACK;  
Rollback terminado.
```

Una **transacción** es una secuencia de una o más sentencias SQL que juntas forman una unidad de trabajo.

### TRANSACCIÓN

Conjunto de sentencias SQL que modifican el estado de la BD (INSERT, UPDATE, DELETE) que deben realizarse de forma atómica (o todas o ninguna)

(Ver tema 7)



## 6. Manipulación de datos. INSERT, UPDATE y DELETE

### 6.5 ROLLBACK, COMMIT y AUTOCOMMIT

La Figura 6.1 muestra transacciones típicas que ilustran las condiciones de COMMIT y ROLLBACK.

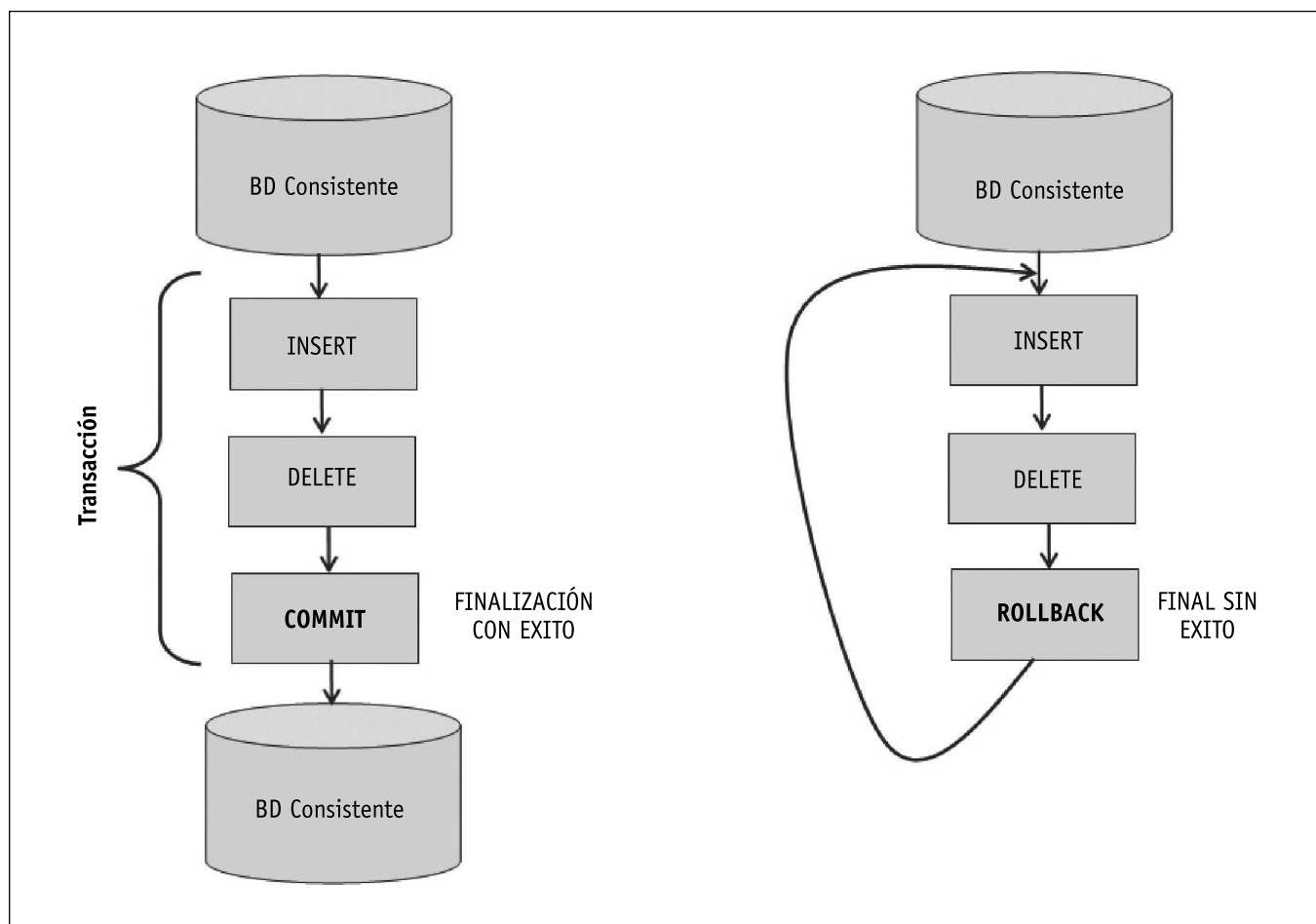


Figura 6.1. Transacciones.



#### Caso práctico

**6** Partimos de la tabla **DEPART** con todas sus filas. El usuario **SCOTT** abre una sesión desde **SQL\*Plus** y ejecuta la orden: **SELECT \* FROM DEPART;** para consultar las filas de la tabla **DEPART**.

- **SCOTT** abre otra sesión ahora desde **iSQL\*Plus**, ejecuta la misma orden para ver el contenido de la tabla. Ambos ven las mismas filas.
- Desde **SQL\*Plus** borra una fila de la tabla **DEPART**: **DELETE DEPART WHERE DEPT\_NO=20;** y consulta de nuevo la tabla. Observa que hay una fila menos.

(Continúa)

## 6. Manipulación de datos. INSERT, UPDATE y DELETE

### 6.5 ROLLBACK, COMMIT y AUTOCOMMIT



(Continuación)

- Desde iSQL\*Plus, consulta el contenido de la tabla DEPART. Observa que se muestran todas las filas.
- Desde SQL\*Plus, ejecuta la orden: COMMIT; Todos los cambios realizados se validan en la base de datos.
- Desde iSQL\*Plus, consulta otra vez el contenido de la tabla DEPART. Observa que ahora no se muestra la fila borrada desde SQL\*Plus.
- Desde SQL\*Plus, ejecuta la orden: DELETE DEPART; después consulta el contenido de la tabla, y observa que no se muestra ninguna fila.
- Se ha confundido al ejecutar la orden porque falta la cláusula WHERE en la sentencia DELETE, entonces ejecuta: ROLLBACK; De nuevo, consulta el contenido de la tabla: se muestran los datos desde el último COMMIT.

#### Actividades propuestas



- 6** Práctica las órdenes ROLLBACK y COMMIT abriendo dos sesiones SQL con el mismo usuario y realiza transacciones sobre tus tablas.

#### A. COMMIT implícito

Hay varias órdenes SQL que fuerzan a que se ejecute un COMMIT sin necesidad de indicarlo:

QUIT	DISCONNECT	CREATE VIEW	ALTER
EXIT	CREATE TABLE	DROP VIEW	REVOKE
CONNECT	DROP TABLE	GRANT	AUDIT
			NOAUDIT

Usar cualquiera de estas órdenes es como usar COMMIT.

#### B. ROLLBACK automático

Si, después de haber realizado cambios en nuestras tablas, se produce un fallo del sistema (por ejemplo, se va la luz) y no hemos validado el trabajo, Oracle hace un ROLLBACK automático sobre cualquier trabajo no validado. Esto significa que tendremos que repetir el trabajo cuando pongamos en marcha la base de datos.