

Gestión y Modelación de Datos

Sesión Practica: Transacciones

MARÍA CONSTANZA PABÓN

MCPABON@JAVERIANACALI.EDU.CO

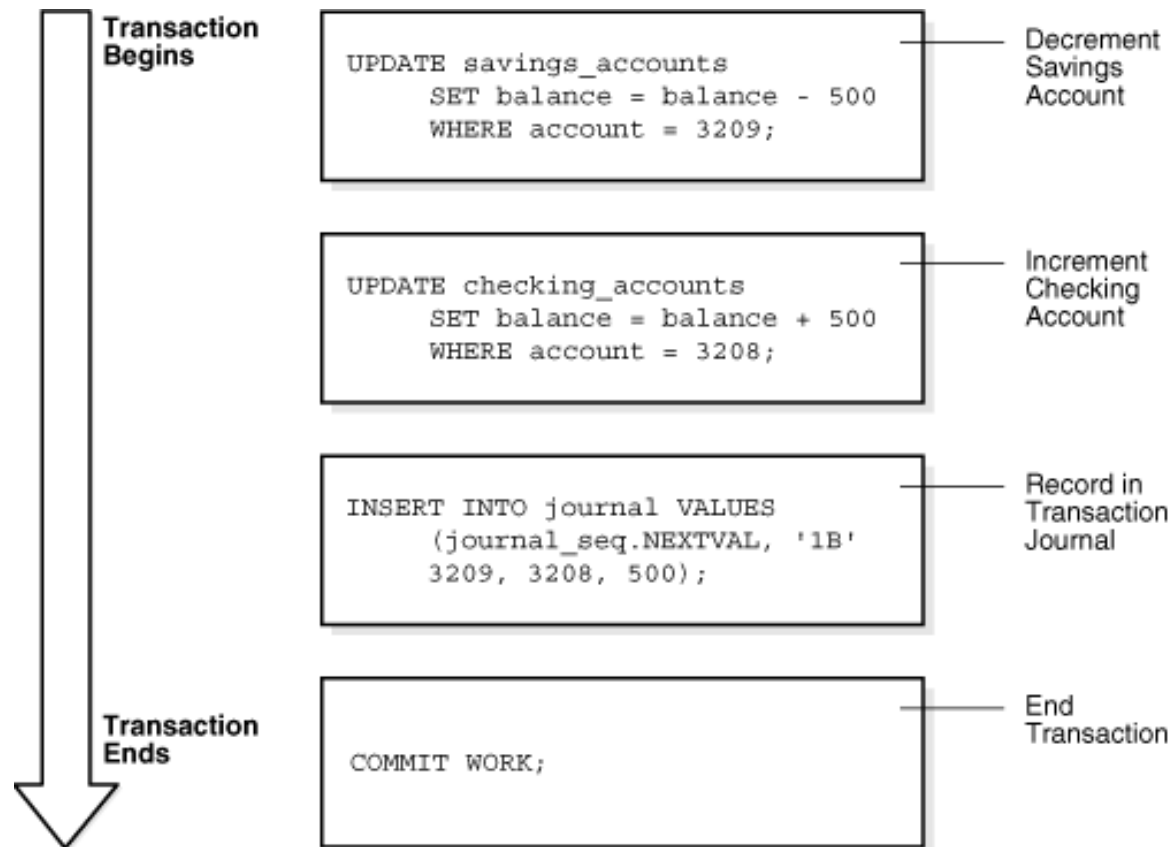


Transacción

Una transacción es una **unidad lógica** de trabajo que contiene una o más instrucciones SQL

- Es una **unidad atómica**
- Todas las sentencias SQL de una transacción pueden ser todas comprometidas (**commit**, aplicadas a la base de datos) o todas retiradas (**rollback**, los cambios se desechan)
- Garantiza el cumplimiento de las **propiedades ACID**

Ejemplo: Transferencia entre cuentas en un banco



Transacción en Oracle

- **Inicia** con la primera sentencia SQL
- **Termina**:
 - Explícitamente: **COMMIT** o **ROLLBACK**
 - Implícitamente: con una **sentencia DDL** o la **terminación** de la aplicación
- Cada sentencia SQL ejecutada es diferente de una transacción comprometida. La ejecución significa que esa sentencia fue:
 - Analizada y se encontró que es una construcción SQL válida
 - Funciona sin errores
 - Pero, mientras no esté comprometida:
 - Los cambios pueden ser reversados
 - Los cambios no son visibles para otros usuarios (bloqueos)

Comandos

Commit

- Hacer **permanentes** los cambios realizados por las sentencias SQL de la transacción

Rollback

- **Deshacer** cualquier cambio en los datos que hayan sido realizados por sentencias SQL de la transacción

Savepoint

- Marcadores intermedios que dividen una transacción en partes más pequeñas

Save points

- Dividen una transacción larga en partes más pequeñas
- Dan la opción de retroceder el trabajo realizado entre el save point y un punto de ejecución
 - Por ejemplo, si un procedimiento contiene varias funciones, se puede crear un save point al inicio de cada función para que en caso de fallo, se devuelvan los cambios que ella realiza
- Internamente, Oracle ubica un savepoint para cada sentencia que ejecuta
 - Si la sentencia falla hace rollback hasta el savepoint

Rollback

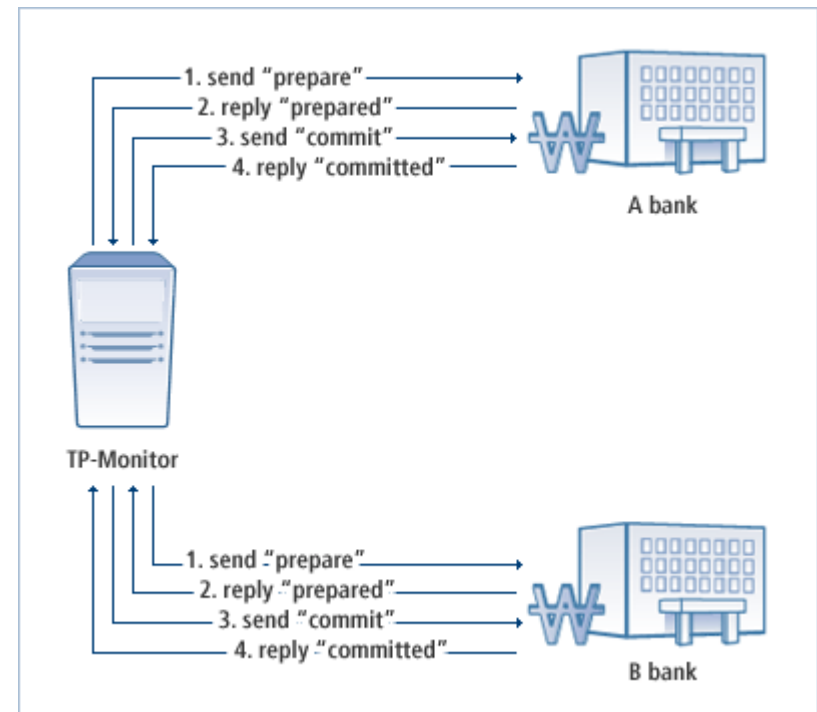
- **A nivel de sentencia:** Si una sentencia SQL genera error o tiene un bloqueo, los cambios que esta sentencia está realizando son reversados
- **Retroceder a un save point**
- **A nivel de transacción** debido a la solicitud (sentencia ROLLBACK) o a la terminación anormal del programa
- De transacciones incompletas: durante una recuperación

Ejemplo

```
CREATE OR REPLACE PROCEDURE ejemplo (per NUMBER) AS
    activo NUMBER(12); pasivo NUMBER(12); invent NUMBER(12);
BEGIN
    SELECT activosCtes, pasivosCtes, inventario INTO activo, pasivo, invent
    FROM balance WHERE periodo = per;
    BEGIN
        SAVEPOINT cero;
        INSERT INTO resumen VALUES ('Liquidez', (activo/pasivo));
        EXCEPTION WHEN OTHERS THEN ROLLBACK TO cero;
    END;
    BEGIN
        SAVEPOINT uno;
        INSERT INTO resumen VALUES ('Capital financiero', (activo - pasivo));
        EXCEPTION WHEN OTHERS THEN ROLLBACK TO uno;
    END;
    BEGIN
        SAVEPOINT dos;
        INSERT INTO resumen VALUES ('Prueba acida', (activo - invent) / pasivo);
        EXCEPTION WHEN OTHERS THEN ROLLBACK TO dos;
    END;
    COMMIT;
END;
```


Mecanismo de dos fases

- **Two phase commit** se aplica en **transacciones distribuidas**
 - Garantiza que todos los servidores que participan en la transacción hacen commit (o todos hacen rollback)
- Es transparente al programa



https://technet.tmaxsoft.com/upload/download/online/tibero/pver-20150504-000002/tibero_dev/ch04.html

Referencias

Oracle 11g. Database Concepts. Transactions.

https://docs.oracle.com/cd/E25054_01/server.1111/e25789/transact.htm