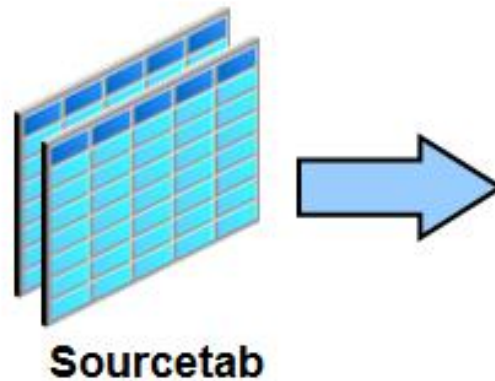


Visión General de Sentencias INSERT de Varias de Tablas

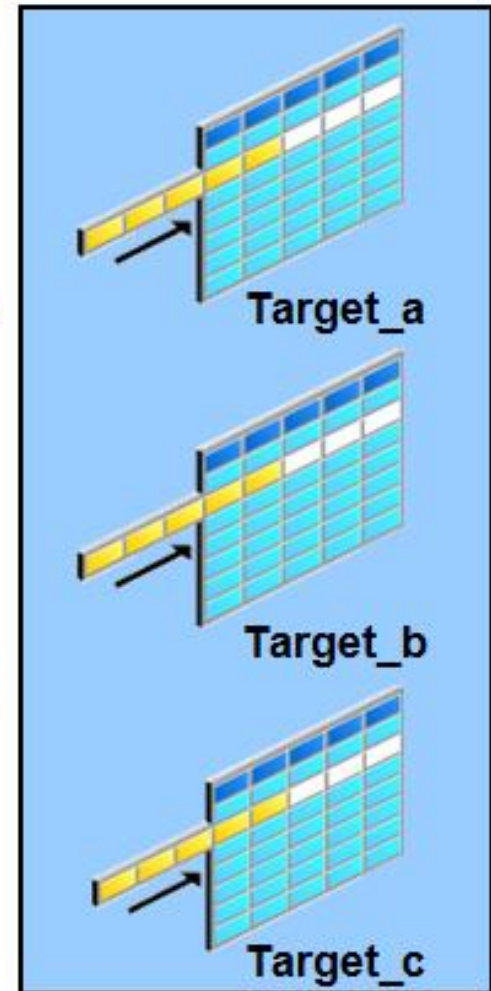
- La sentencia **INSERT de varias tablas** es útil en escenarios de almacén de datos en los que se debe cargar datos regularmente para facilitar el análisis de negocio. Para eso se realiza un proceso de extraer datos y copiarlos en el almacén (ETL)
- Los datos pueden extraerse de múltiples orígenes y luego deben almacenarse en algún sistema intermedio para su posterior procesamiento. Después de cargar los datos en la base de datos, se pueden ejecutar transformaciones mediante operaciones SQL
- La sentencia INSERT de varias tablas es una de las técnicas utilizadas para implantar las transformaciones de datos SQL

Visión General de Sentencias INSERT de Varias de Tablas



```
INSERT ALL  
  INTO target_a VALUES (...,...,...)  
  INTO target_b VALUES (...,...,...)  
  INTO target_c VALUES (...,...,...)  
SELECT ...  
FROM sourcetab  
WHERE ...;
```

Subconsulta



Visión General de Sentencias INSERT de Varias de Tablas

- También puede utilizar la sentencia INSERT...SELECT para insertar filas en varias tablas como parte de una única sentencia DML.
- Las sentencias INSERT de varias tablas se utilizan en sistemas de almacenamiento de datos para transferir datos de uno o más orígenes operativos a un juego de tablas de destino.
- Proporcionan una mejora significativa del rendimiento en:
 - Sentencia única DML frente a diversas sentencias INSERT...SELECT
 - Una única sentencia DML frente a un procedimiento para realizar varias inserciones mediante la sintaxis IF...THEN

Tipos de Sentencias INSERT de Varias Tablas

- Los diferentes tipos de sentencias INSERT de varias tablas son:
 - **INSERT incondicional:** por cada fila devuelta por la subconsulta, se inserta una fila en cada una de las tablas de destino
 - **INSERT ALL condicional:** por cada fila devuelta por la subconsulta, se inserta una fila en cada una de las tablas de destino si se cumple la condición especificada
 - **INSERT FIRST condicional:** por cada fila devuelta por la subconsulta, se inserta una fila en la primera tabla de destino en la que se cumple la condición
 - **INSERT mediante giro:** Es un caso especial del insert condicional

INSERT ALL Incondicional

```
CREATE TABLE CLIENTES_CANTIDADES (ID_CLIENTE NUMBER,  
CANTIDAD NUMBER);  
CREATE TABLE CLIENTES_MONTOS(ID_CLIENTE NUMBER,  
MONTO NUMBER);
```

INSERT ALL

```
  INTO clientes_cantidades VALUES (id_cliente, cantidad)  
  INTO clientes_montos VALUES (id_cliente, monto_total)  
SELECT id_cliente, count(*) cantidad, sum(monto_total)  
monto_total  
FROM b_ventas  
group by id_cliente;
```

INSERT ALL Incondicional

- Se denomina “incondicional” porque no se aplican más restricciones a las filas que las impuestas por la condición del select.
- Todas las filas recuperadas se insertan en ambas tablas
- La clausula VALUES en las sentencias insert especifican las columnas del select que se deben insertar en cada tabla
- Cada fila devuelta por el select da como resultado dos inserciones: una en clientes_cantidades, otra en clientes_montos

INSERT ALL Condicional

```
CREATE TABLE CLIENTES_MAYORISTAS (ID_CLIENTE NUMBER, MONTO NUMBER);  
CREATE TABLE CLIENTES_IDEN (ID_CLIENTE NUMBER, MONTO NUMBER);  
CREATE TABLE CLIENTES_MINORISTAS (ID_CLIENTE NUMBER, MONTO NUMBER);
```

INSERT ALL

```
WHEN monto_total < 1000000 THEN  
  INTO clientes_MINORISTAS
```

Dependiendo del monto de venta,
inserta en clientes_minoristas o
clientes_mayoristas

```
WHEN monto_total >= 1000000 THEN  
  INTO clientes_MAYORISTAS
```

```
WHEN id_cliente < 10 THEN  
  INTO clientes_IDEN
```

Si además el código es menor a 10,
inserta también en clientes_iden

```
SELECT id_cliente, sum(monto_total) monto_total  
  FROM b_ventas  
 group by id_cliente;
```

INSERT ALL Condicional

- A las filas se aplica solo la condición de recuperación del SELECT
- Las filas se insertan en las tablas respectivas solo si cumplen la condición WHEN. Es posible que la misma fila se inserte en solo una o en varias tablas, dependiendo de si satisfacen o no dicha condición

INSERT FIRST Condicional

INSERT FIRST

WHEN id_cliente <10 THEN

INTO clientes_IDEN

WHEN monto_total < 1000000 THEN

INTO clientes_MINORISTAS

WHEN monto_total >= 1000000 THEN

INTO clientes_MAYORISTAS

SELECT id_cliente, sum(monto_total) monto_total

FROM b_ventas

group by id_cliente;

Inserta en la tabla con la primera
condicion WHEN que se cumpla

INSERT FIRST Condicional

- Las filas recuperadas por el select se insertan en la primera tabla en la que se cumpla la condición WHEN. Cada fila recuperada por el select se inserta solo en una tabla.

INSERT mediante giro

- Convertir el juego de registros de ventas de la tabla de base de datos no relacional a un formato relacional.

Emp_ID	Week_ID	MON	TUES	WED	THUR	FRI
176	6	2000	3000	4000	5000	6000



Employee_ID	WEEK	SALES
176	6	2000
176	6	3000
176	6	4000
176	6	5000
176	6	6000

INSERT mediante giro o pivotante

```
INSERT ALL
  INTO sales_info VALUES (employee_id, week_id, sales_MON)
  INTO sales_info VALUES (employee_id, week_id, sales_TUE)
  INTO sales_info VALUES (employee_id, week_id, sales_WED)
  INTO sales_info VALUES (employee_id, week_id, sales_THUR)
  INTO sales_info VALUES (employee_id, week_id, sales_FRI)
SELECT EMPLOYEE_ID, week_id, sales_MON, sales_TUE,
       sales_WED, sales_THUR, sales_FRI
FROM sales_source_data;
```

- El giro es una operación de transformación de forma que cada registro de cualquier flujo de entrada, como una tabla no relacional, se convierta en registros para un entorno relacional.