

UNIVERSIDAD NACIONAL DE CHIMBORAZO
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
TALLER PRÁCTICO ETL – TERCERA PARTE

Profesora: Ing. Elba Boderó Poveda, Ph.D.



PRERREQUISITOS:

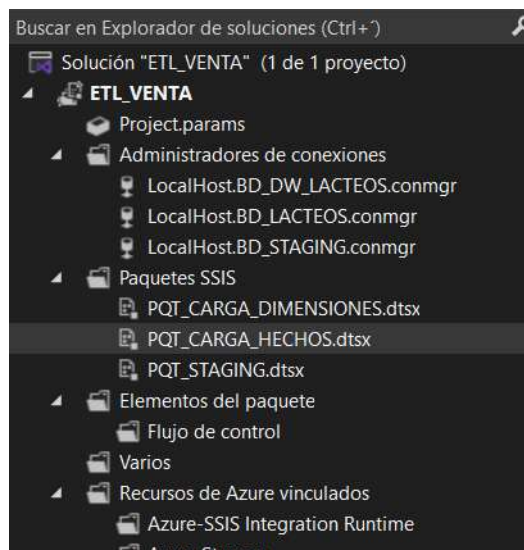
- SQL Server
- SQL server Data Tools en Visual Studio.
- Base de datos Fuente.
- Base de datos para Staging creada.
- Base de datos del Data Warehouse creada:
 - Utilizar el diseño del **MODELO DIMENSIONAL** de la DB.
 - Ejecutar el store procedure carga dimensión tiempo.

INSTRUCCIONES:

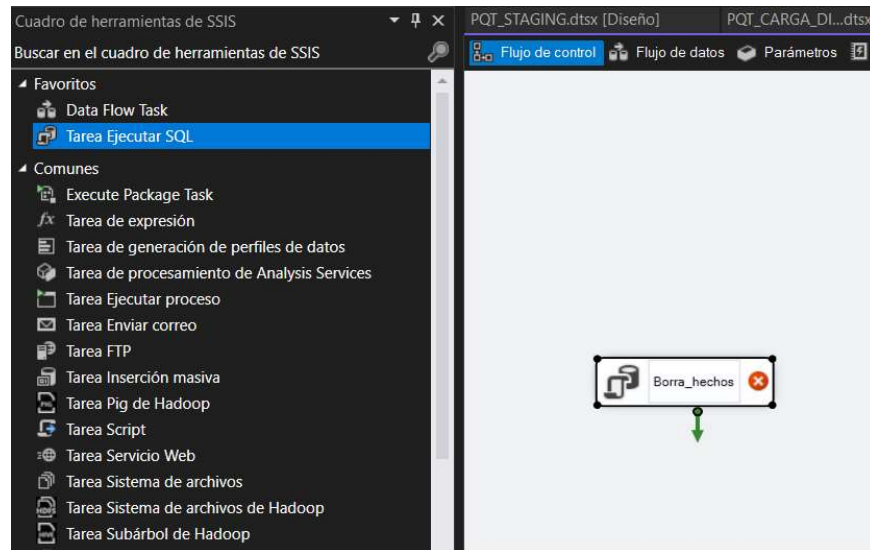
1. Una vez generado el paquete de CARGA A STAGING y el paquete de CARGA A DIMENSIONES.

PARTE 3: CARGA DE HECHOS

1. Abra el paquete PQT_CARGA_HECHOS creado.



2. Cree una “Tarea Ejecutar SQL” y denomínela “Borra_hechos”.

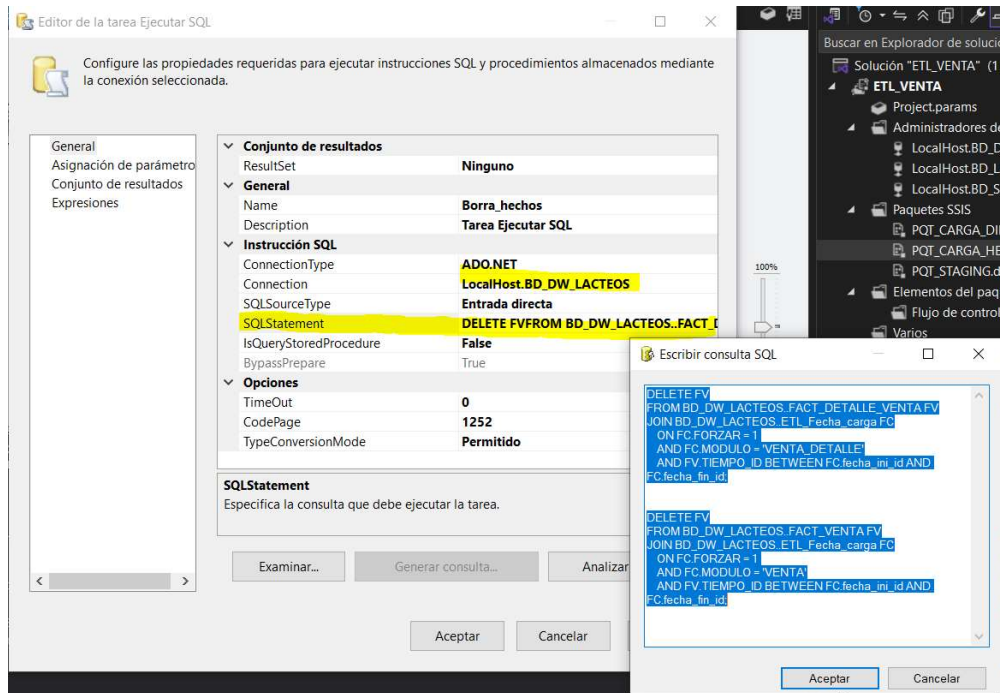


En la opción “SQLStatement” de “Borra_hechos” colocar lo siguiente:

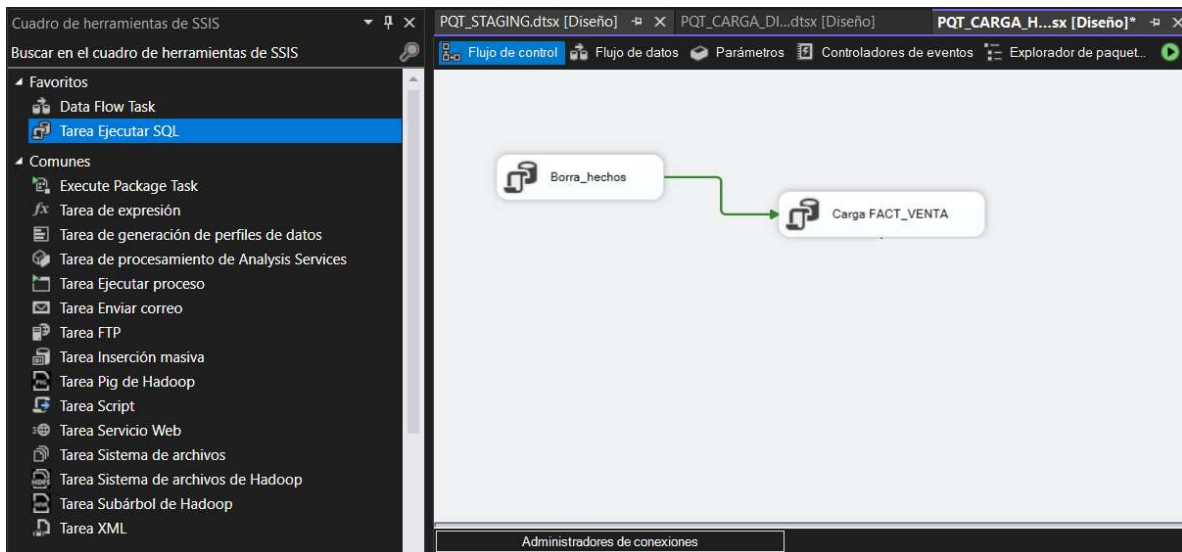
```
DELETE FV
FROM BD_DW_LACTEOS..FACT_DETALLE_VENTA FV
JOIN BD_DW_LACTEOS..ETL_Fecha_carga FC
  ON FC.FORZAR = 1
  AND FC.MODULO = 'VENTA_DETALLE'
  AND FV.TIEMPO_ID BETWEEN FC.fecha_ini_id AND FC.fecha_fin_id;
```

```
DELETE FV
FROM BD_DW_LACTEOS..FACT_VENTA FV
JOIN BD_DW_LACTEOS..ETL_Fecha_carga FC
  ON FC.FORZAR = 1
  AND FC.MODULO = 'VENTA'
  AND FV.TIEMPO_ID BETWEEN FC.fecha_ini_id AND FC.fecha_fin_id;
```

Estas dos sentencias eliminan registros **por rangos de fechas identificados en la tabla de control ETL_Fecha_carga**, siempre y cuando el proceso esté marcado como FORZAR = 1. Esta estrategia permite controlar de forma precisa la eliminación de datos antes de una recarga parcial de información en el Data Warehouse, garantizando que no existan duplicados ni inconsistencias. Es una **práctica común y recomendada** en entornos de inteligencia de negocios para mantener la integridad y actualidad del sistema analítico.



3. Cree una “Tarea Ejecutar SQL” y denomínela “Carga FACT_VENTA”.



En la opción “SQLStatement” de “Carga FACT_VENTA” colocar lo siguiente:

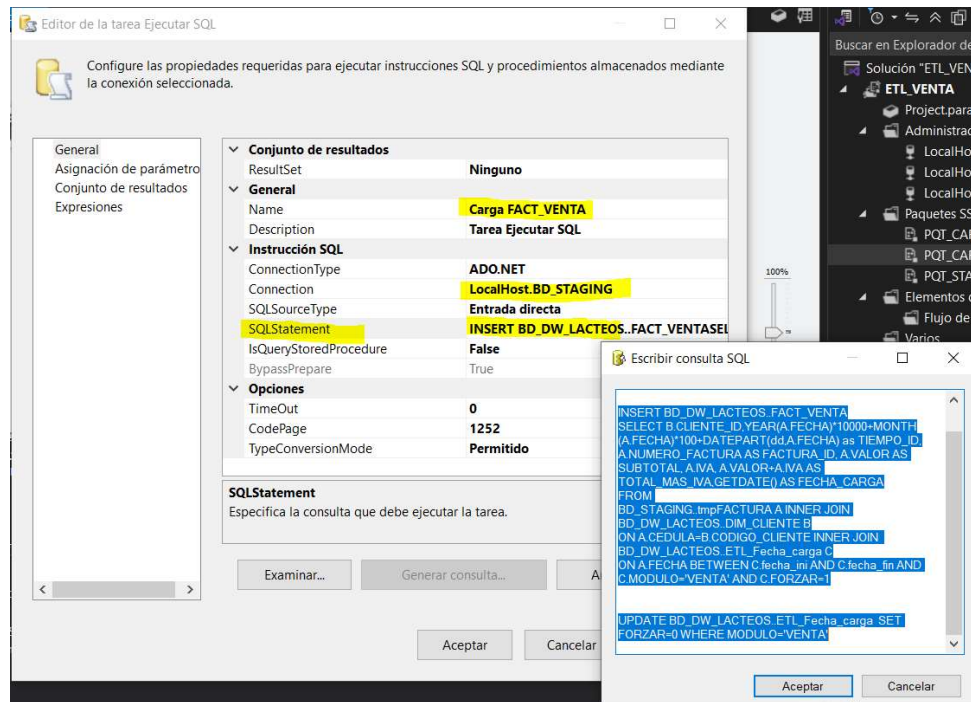
```

INSERT BD_DW_LACTEOS..FACT_VENTA
SELECT B.CLIENTE_ID, YEAR(A.FECHA)*10000+MONTH(A.FECHA)*100+DATEPART(dd,A.FECHA) as
TIEMPO_ID,
A.NUMERO_FACTURA AS FACTURA_ID, A.VALOR AS SUBTOTAL, A.IVA, A.VALOR+A.IVA AS
TOTAL_MAS_IVA, GETDATE() AS FECHA_CARGA
FROM
BD_STAGING..tmpFACTURA A INNER JOIN BD_DW_LACTEOS..DIM_CLIENTE B
ON A.CEDULA=B.CODIGO_CLIENTE INNER JOIN BD_DW_LACTEOS..ETL_Fecha_carga C
ON A.FECHA BETWEEN C.fecha_ini AND C.fecha_fin AND C.MODULO='VENTA' AND C.FORZAR=1

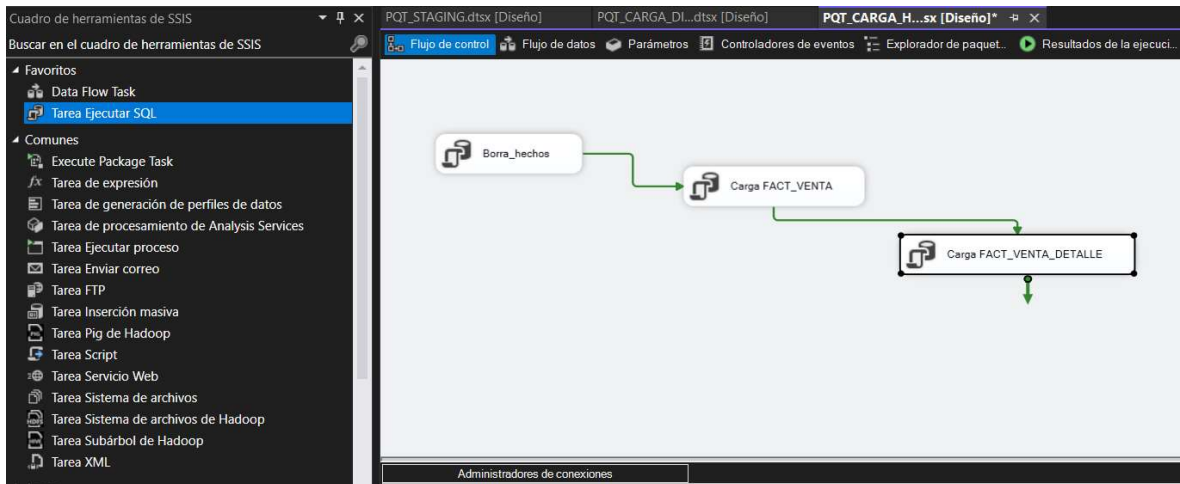
UPDATE BD_DW_LACTEOS..ETL_Fecha_carga SET FORZAR=0 WHERE MODULO='VENTA'

```

Este bloque de código implementa la **carga controlada y segura de datos desde staging hacia el Data Warehouse** en el contexto de un módulo de ventas. La lógica está diseñada para funcionar **bajo un modelo de recarga forzada**: solo se carga si el control FORZAR = 1 está activo, y al finalizar, se actualiza dicha bandera para prevenir duplicidades.



4. Cree una "Tarea Ejecutar SQL" y denomínela "Carga FACT_VENTA_DETALLE".



En la opción “SQLStatement” de “Carga FACT_VENTA_DETALLE” colocar lo siguiente:

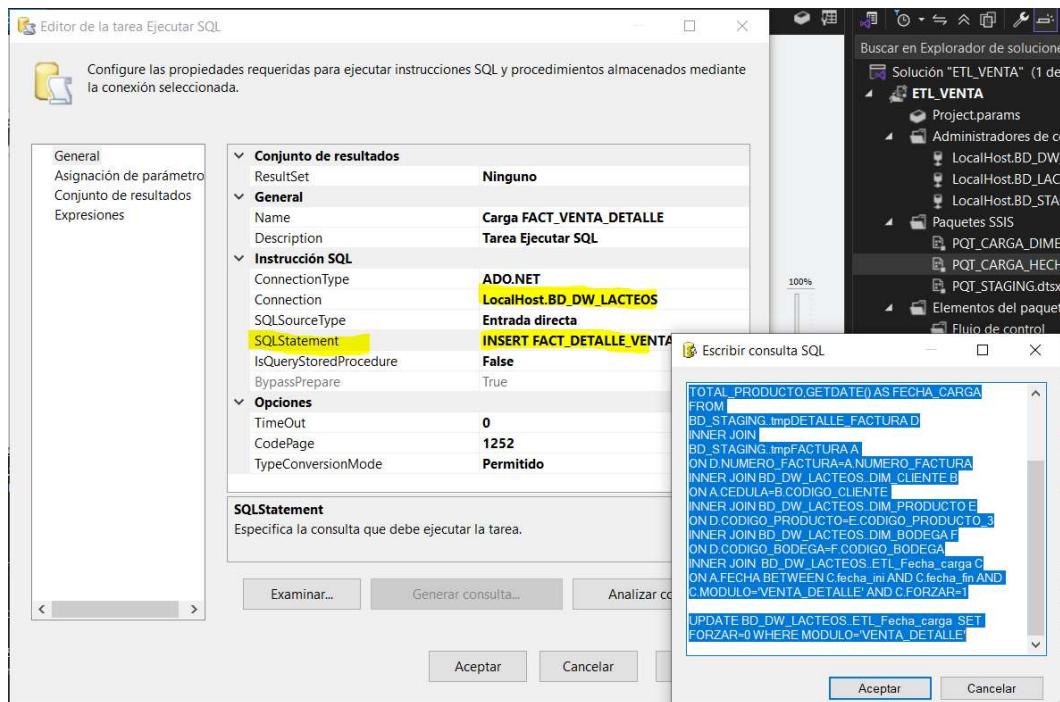
```

INSERT FACT_DETALLE_VENTA
SELECT E.PRODUCTO_ID,F.BODEGA_ID,
YEAR(A.FECHA)*10000+MONTH(A.FECHA)*100+DATEPART(dd,A.FECHA) as TIEMPO_ID,B.CLIENTE_ID,
A.NUMERO_FACTURA AS FACTURA_ID, D.CANTIDAD,D.VALOR_UNITARIO,D.TOTAL AS
TOTAL_PRODUCTO,GETDATE() AS FECHA_CARGA
FROM
BD_STAGING..tmpDETALLE_FACTURA D
INNER JOIN
BD_STAGING..tmpFACTURA A
ON D.NUMERO_FACTURA=A.NUMERO_FACTURA
INNER JOIN BD_DW_LACTEOS..DIM_CLIENTE B
ON A.CEDULA=B.CODIGO_CLIENTE
INNER JOIN BD_DW_LACTEOS..DIM_PRODUCTO E
ON D.CODIGO_PRODUCTO=E.CODIGO_PRODUCTO_3
INNER JOIN BD_DW_LACTEOS..DIM_BODEGA F
ON D.CODIGO_BODEGA=F.CODIGO_BODEGA
INNER JOIN BD_DW_LACTEOS..ETL_Fecha_carga C
ON A.FECHA BETWEEN C.fecha_ini AND C.fecha_fin AND C.MODULO='VENTA_DETALLE' AND C.FORZAR=1

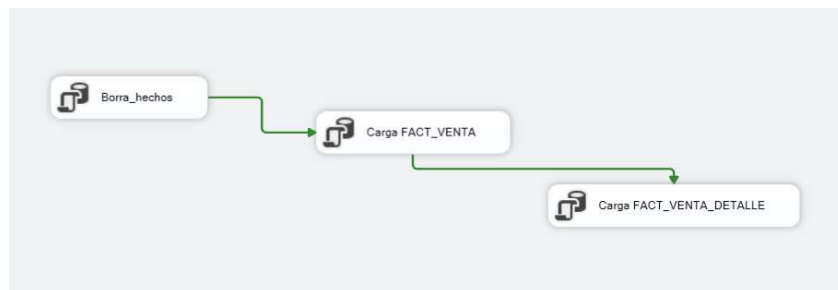
UPDATE BD_DW_LACTEOS..ETL_Fecha_carga SET FORZAR=0 WHERE MODULO='VENTA_DETALLE'

```

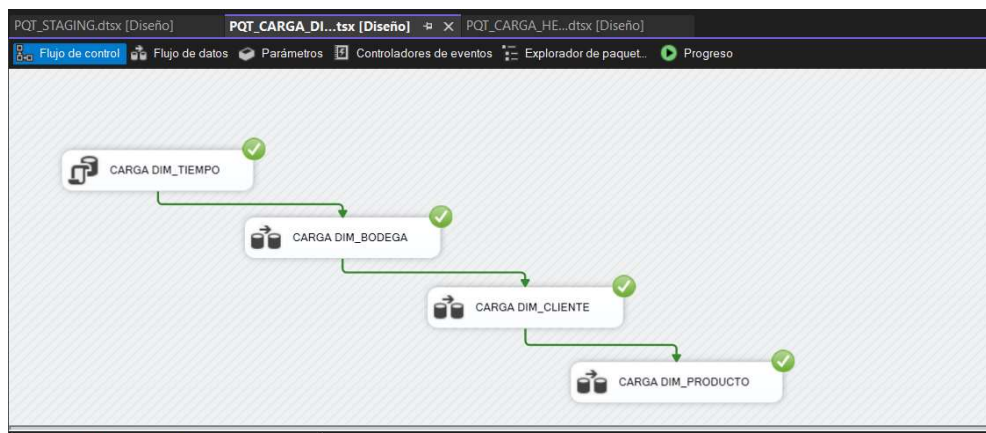
Este bloque de código implementa la **carga detallada de datos transaccionales de ventas** desde el staging hacia la tabla de hechos FACT_DETALLE_VENTA, de forma controlada y segura. A través de la lógica de control basada en la tabla ETL_Fecha_carga, permite gestionar cargas incrementales o forzadas, asegurando precisión, trazabilidad y eficiencia en el flujo ETL de una solución de BI.



5. Finalmente, nuestro paquete con sus flujos de hechos debe mostrarse de la siguiente forma:



6. Ahora puede ejecutar el paquete de Carga de Dimensiones:



7. Ahora puede ejecutar el paquete de Carga de Hechos:

