

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



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

**PRERREQUISITOS:**

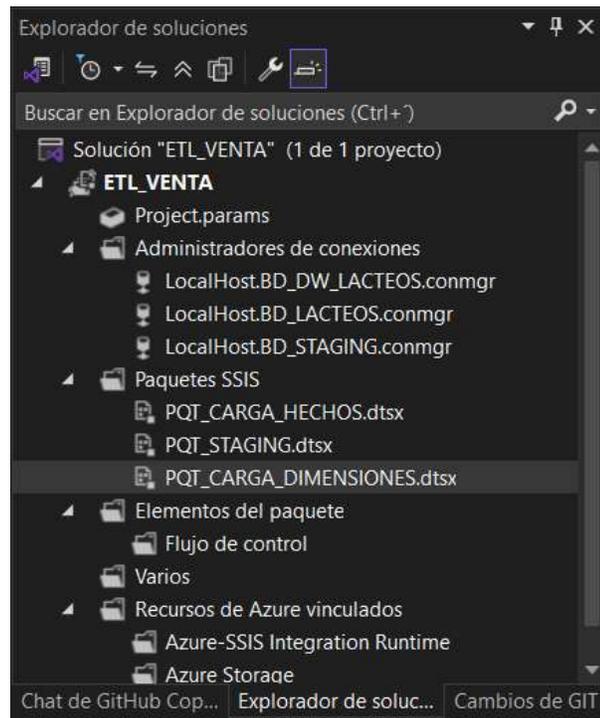
- SQL Server
- SQL server Data Tools en Visual Studio.
- Base de datos Fuente.
- Base de datos para Staging creada.
- Base de datos del Data Warehouse creada:
  - Utilizar el diseño del **MODELO DIMENSIONAL** de la DB.
  - Ejecutar el store procedure carga dimensión tiempo.

**INSTRUCCIONES:**

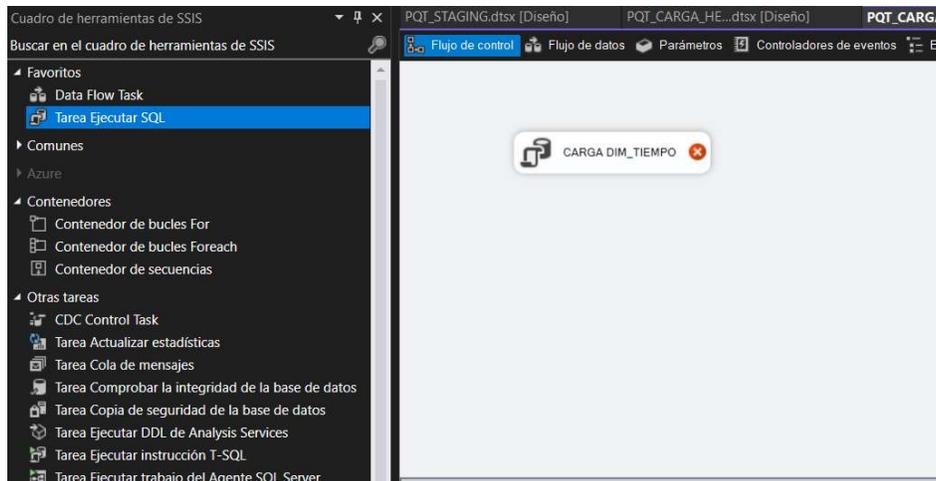
1. Una vez generado el paquete de CARGA A STAGING.

**PARTE 2: CARGA DIMENSIONES**

1. Abra el paquete PQT\_CARGA\_DIMENSIONES creado.



2. Cree una “Tarea Ejecutar SQL” y denomínela “CARGA DIM\_TIEMPO”.



En la opción "SQLStatement" de "CARGA DIM\_TIEMPO" colocar lo siguiente:

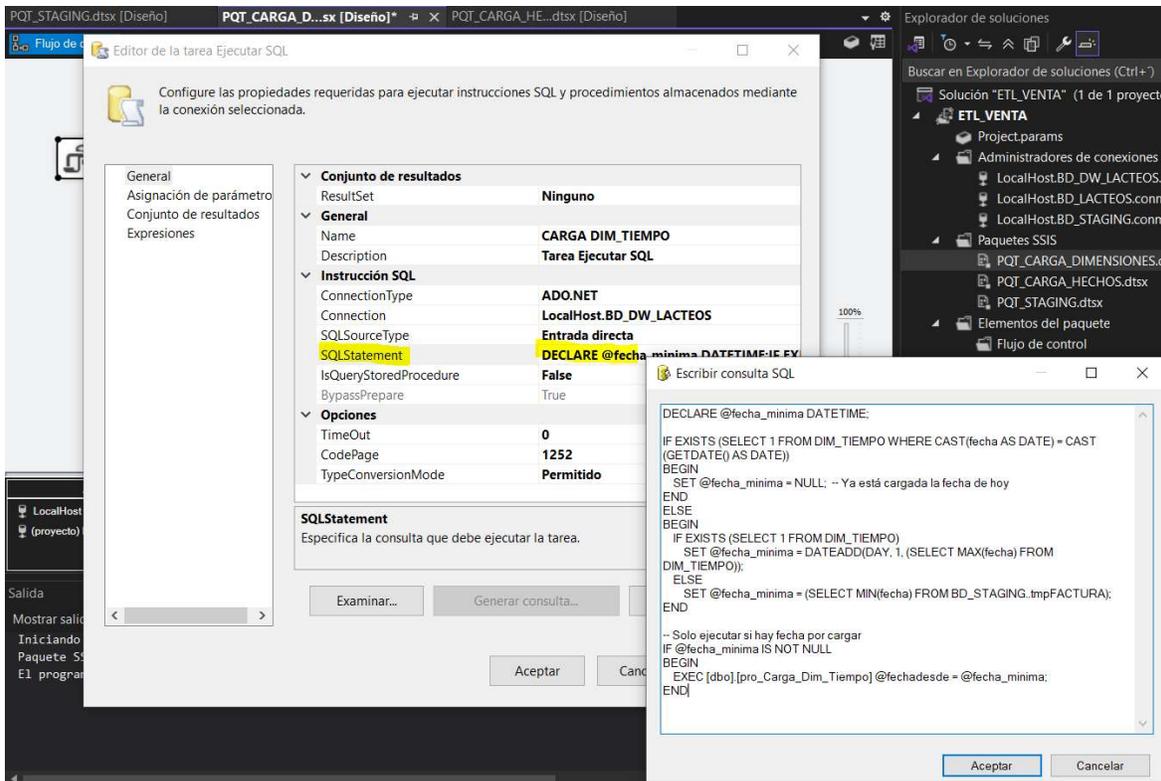
```

DECLARE @fecha_minima DATETIME;
IF EXISTS (SELECT 1 FROM DIM_TIEMPO WHERE CAST(fecha AS DATE) = CAST(GETDATE() AS DATE))
BEGIN
    SET @fecha_minima = NULL;
END
ELSE
BEGIN
    IF EXISTS (SELECT 1 FROM DIM_TIEMPO)
        SET @fecha_minima = DATEADD(DAY, 1, (SELECT MAX(fecha) FROM DIM_TIEMPO));
    ELSE
        SET @fecha_minima = (SELECT MIN(fecha) FROM BD_STAGING..tmpFACTURA);
END
IF @fecha_minima IS NOT NULL
BEGIN
    EXEC [dbo].[pro_Carga_Dim_Tiempo] @fechadesde = @fecha_minima;
END

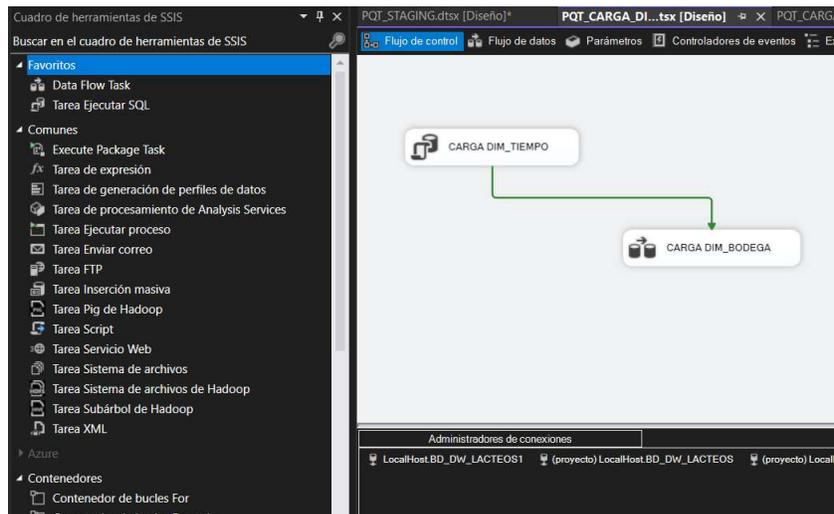
```

Este bloque dinámicamente calcula la fecha mínima de los datos temporales (tmpFACTURA) y la utiliza como fecha de inicio (@fechadesde) para cargar o actualizar la dimensión de tiempo (Dim\_Tiempo), a través del procedimiento pro\_Carga\_Dim\_Tiempo.

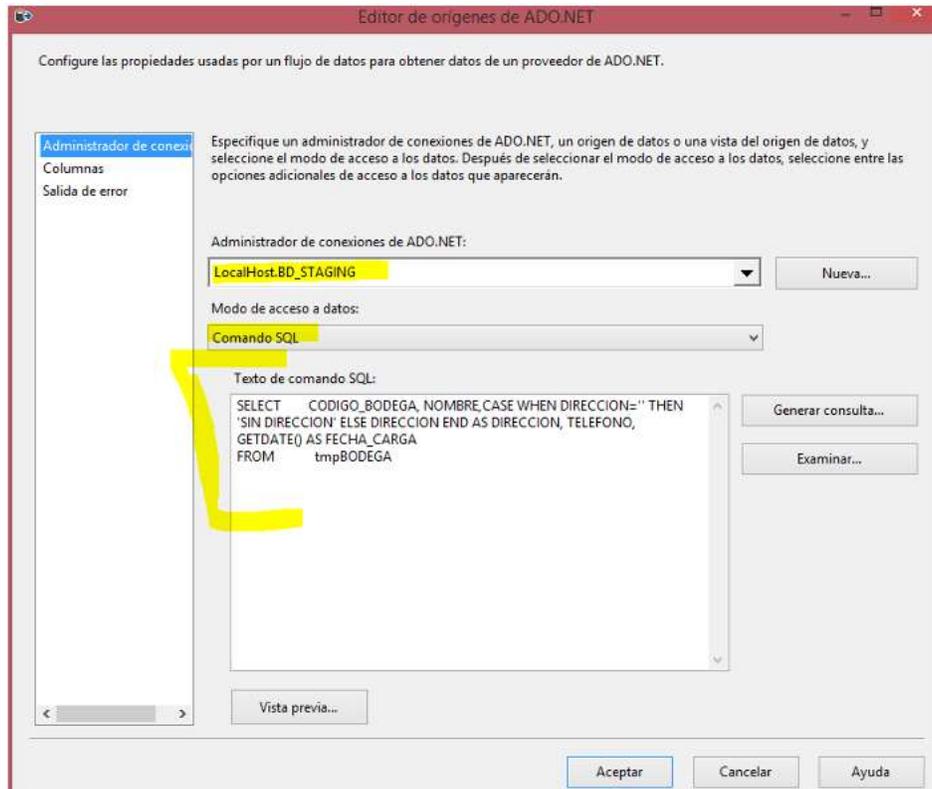
Es útil en entornos ETL donde se quiere asegurar que solo se cargue la parte relevante del tiempo en la dimensión, minimizando procesamiento redundante.



3. Para cargar DIM\_BODEGA cree una tarea de flujo de datos.

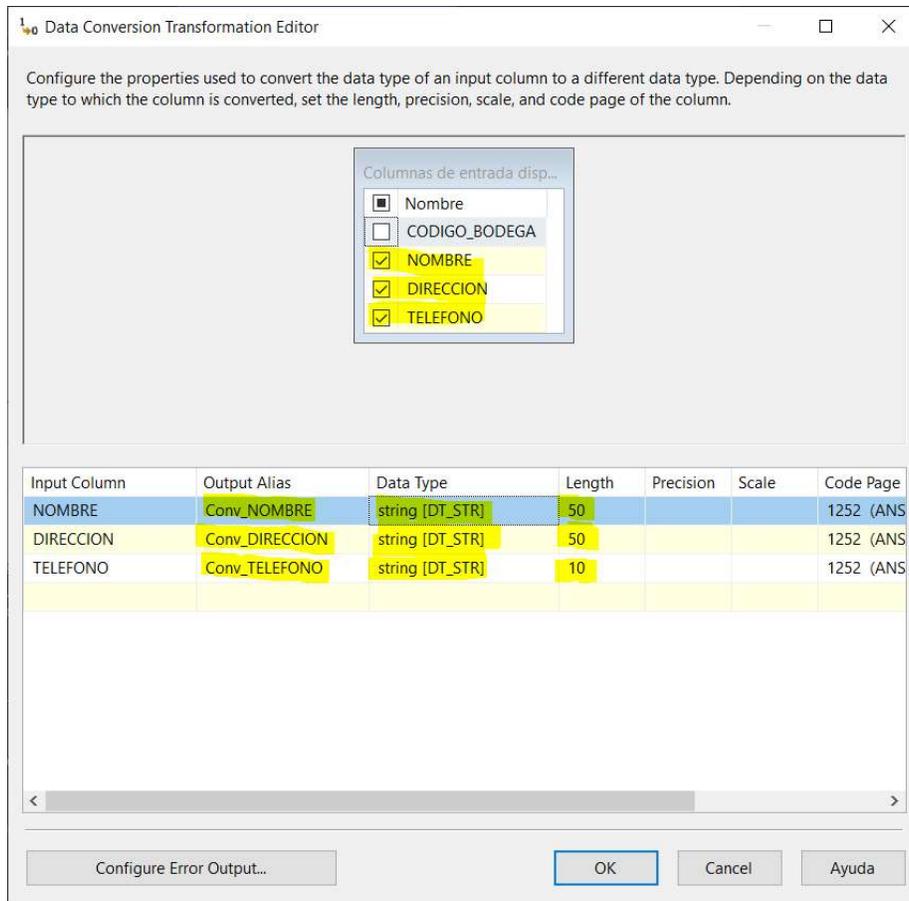


4. Doble clic en la tarea de flujo de datos creada y cree un origen de datos de ADO.NET que haga referencia a la tabla BODEGA.

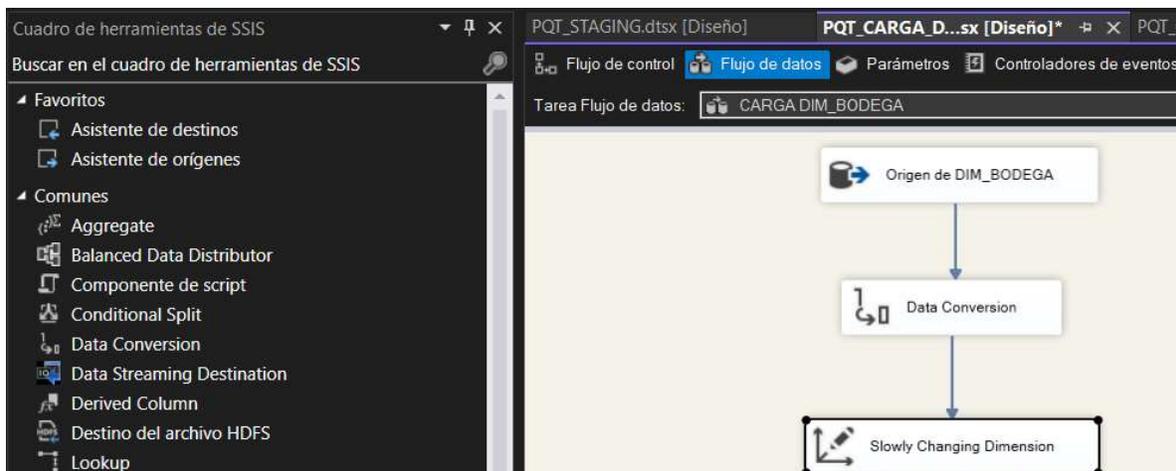


5. Añada una herramienta “Conversión de datos” y configúrela. La Conversión de Datos, es una herramienta muy importante debido a que las fuentes de datos suelen poseer una codificación diferente a la de destino o a su vez otros tipos de datos.





6. Agregue una herramienta de “Dimensión de variación lenta”.



En SQL Server Integration Services (SSIS), la herramienta conocida como “Dimensión de variación lenta” se utiliza para manejar los cambios en los datos de las dimensiones de un Data Warehouse, donde se requiere mantener el historial de estos cambios o actualizarlos según el tipo de cambio.

7. Configure la herramienta “Dimensión de variación lenta” de la siguiente manera:

Asistente para dimensiones de variación lenta

**Seleccionar una tabla de dimensiones y claves**  
Seleccione una tabla de dimensiones para cargar y asignar columnas en la entrada de transformación para columnas de la tabla de dimensiones.

Administrador de conexiones:  
LocalHost.BD\_DW\_LACTEOS1 Nueva...

Tabla o vista:  
[dbo].[DIM\_BODEGA]

Columnas de entrada	Columnas de dimensión	Tipo de clave
CODIGO_BODEGA	CODIGO_BODEGA	Clave empresarial
DIRECCION	DIRECCION	No es una columna de ...
FECHA_CARGA	FECHA_CARGA	No es una columna de ...
NOMBRE	NOMBRE	No es una columna de ...
TELEFONO	TELEFONO	No es una columna de ...

Help < Back Next > Finish >> Cancel

Asistente para dimensiones de variación lenta

**Columnas de dimensión de variación lenta**  
Administre los cambios en los datos de columna en las dimensiones de variación lenta mediante la definición del tipo de cambio para las columnas de dimensión.

**Atributo fijo**  
Seleccione este tipo cuando el valor de una columna no deba cambiar. Los cambios se tratan como errores.

**Atributo variable**  
Seleccione este tipo cuando los valores cambiados deban sobrescribir los valores existentes. Este es un

**Atributo histórico**  
Seleccione este tipo cuando los cambios en los valores de columna se guarden en registros nuevos. Los valores anteriores se guardan en registros marcados como no actualizados. Este cambio es de Tipo 2.

Seleccione un tipo de cambio para columnas de dimensión de variación lenta:

Columnas de dimensión	Cambiar tipo
DIRECCION	Atributo variable
NOMBRE	Atributo variable
TELEFONO	Atributo variable

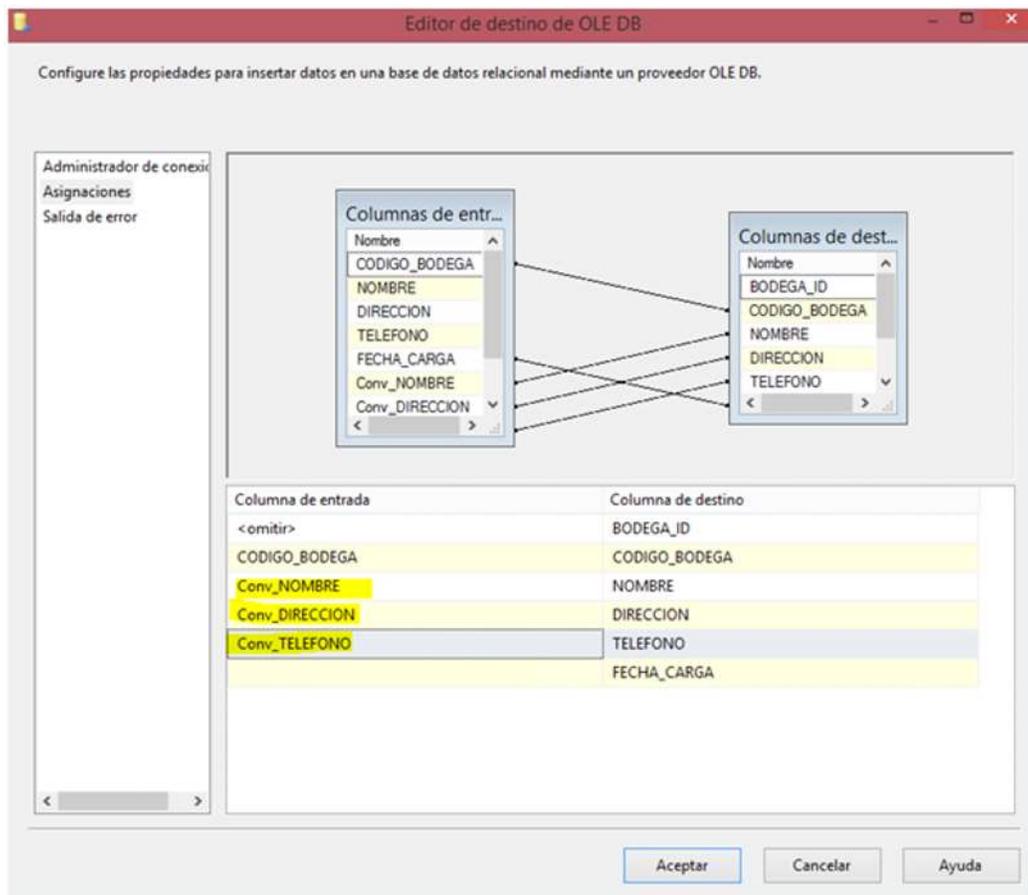
Quitar

Help < Back Next > Finish >> Cancel

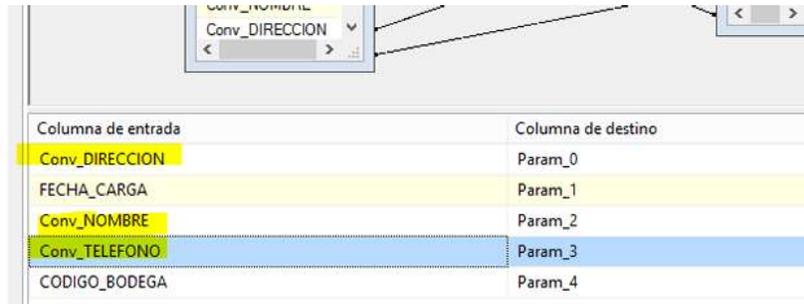
Se asume que los atributos cuando sean modificados en las bases fuentes podrán ser sobrescritos, en adelante pulse next hasta finalizar.



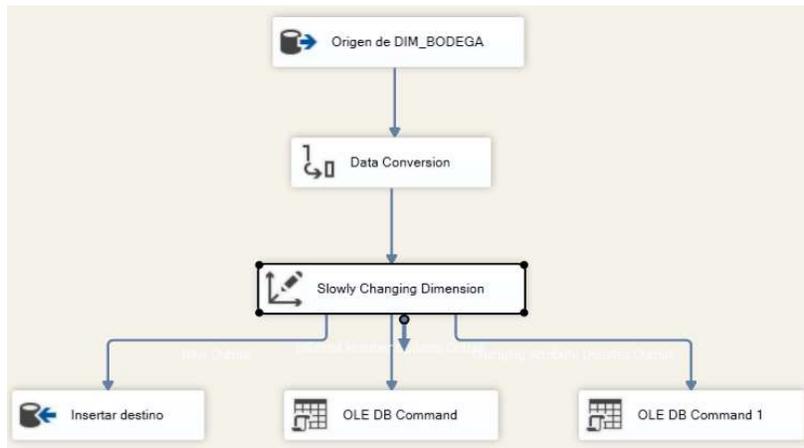
8. Configure "Insertar destino" de la herramienta de "Dimensión de variación lenta":



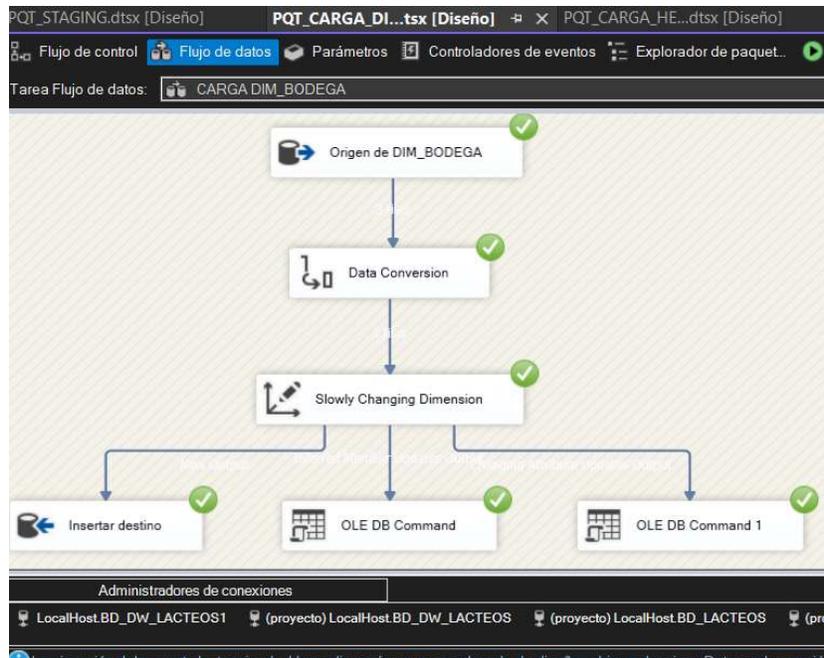
9. De la misma manera configure “Comando de OLE DB” y “Comando de OLE DB 1” de la herramienta de “Dimensión de variación lenta”:



Columna de entrada	Columna de destino
Conv_DIRECCION	Param_0
FECHA_CARGA	Param_1
Conv_NOMBRE	Param_2
Conv_TELEFONO	Param_3
CODIGO_BODEGA	Param_4



10. Ejecute tarea para comprobar que funcione correctamente:



11. El resultado de la ejecución en el DW es:

```

1 SELECT TOP (1000) [BODEGA_ID]
2     , [CODIGO_BODEGA]
3     , [NOMBRE]
4     , [DIRECCION]
5     , [TELEFONO]
6     , [FECHA_CARGA]
7 FROM [BD_DW_LACTEOS].[dbo].[DIM_BODEGA]
8

```

	BODEGA_ID	CODIGO_BODEGA	NOMBRE	DIRECCION	TELEFONO	FECHA_CARGA
1	1	1	CENTRO	AV GALO PLAZA	2123456	2025-05-25 12:37:25.720
2	2	2	NORTE	AV GALO PLAZA	2345678	2025-05-25 12:37:25.720
3	3	3	SUR	AV MALDONADO	2314768	2025-05-25 12:37:25.720

12. Realice el proceso con las demás dimensiones, al final nuestro flujo debe mostrarse de la siguiente manera:



13. En la CARGA DIM\_PRODUCTO, elija el modo de acceso a datos con COMANDO SQL y agregue:

```

select DISTINCT P1.CODIGO_PRODUCTO as CODIGO_PRODUCTO_1,P1.DESCRIPCION as DESCRIPCION_1,
isnull(P2.CODIGO_PRODUCTO,P1.CODIGO_PRODUCTO) as
CODIGO_PRODUCTO_2,isnull(P2.DESCRIPCION,P1.DESCRIPCION) as DESCRIPCION_2,
isnull(P3.CODIGO_PRODUCTO,isnull(P2.CODIGO_PRODUCTO,P1.CODIGO_PRODUCTO)) as
CODIGO_PRODUCTO_3,
isnull(P3.DESCRIPCION,isnull(P2.DESCRIPCION,P1.DESCRIPCION)) as
DESCRIPCION_3,ISNULL(P3.PRECIO,ISNULL(P2.PRECIO,P1.PRECIO)) AS PRECIO
,GETDATE() AS
FECHA_CARGA
from tmpPRODUCTO P1
left join tmpPRODUCTO P2
on P1.CODIGO_PRODUCTO= P2.CODIGO_PADRE
left join tmpPRODUCTO P3
on P2.CODIGO_PRODUCTO= P3.CODIGO_PADRE
WHERE
P1.CODIGO_PADRE="" OR P1.CODIGO_PADRE IS NULL

```

Esta consulta SQL está diseñada para **construir una jerarquía de productos padre-hijo** a partir de una tabla temporal tmpPRODUCTO que contiene relaciones recursivas entre productos y sus derivados o presentaciones, considerando hasta tres niveles de profundidad. Este tipo de estructura es común en catálogos de productos con versiones, paquetes o agrupaciones.

El propósito de la consulta es construir una vista jerárquica de productos a partir de relaciones padre-hijo en la tabla tmpPRODUCTO, consolidando información relevante como descripción y precio desde los niveles inferiores hacia el superior. Esta estructura puede servir como base para poblar una dimensión de productos (DIM\_PRODUCTO) en un Data Warehouse.

