

BASES DE DATOS II

PREICA2501B010095

S30 -EVIDENCIA DE APRENDIZAJE 3

PROCESO DE TRANSFORMACIÓN DE DATOS Y CARGA EN EL DATA MART FINAL

REALIZADO POR:

Grupo:

BASEDEDATOSIIPRE 5

JEREMY IVAN PEDRAZA HERNANDEZ

PRESENTADO A:

INSTRUCTOR

VICTOR HUGO MERCADO RAMOS

INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA

2025

Contenido

Planteamiento del Problema	3
Introducción	4
Objetivos	4
Análisis de la base de datos (Problema)	4
Identificación de campos necesarios	6
Diseño del modelo estrella	6
Representación Gráfica del modelo estrella	7
Construcción de tablas desde herramienta SSIS.....	8
Crear tabla de dimensiones.....	9
Tabla dimensión cliente	9
Tabla Dimensión producto	10
Tabla dimensión tiempo	11
Tabla dimensión empleado	12
Tabla dimensión oficina	13
Tabla de hechos ventas	14
Ejecutamos la tarea y validamos los resultados	15
Flujo de Data Flow Task	16
Cargue de Dimensión oficina	17
Cargue dimensión empleado	21
Cargue Dimensión Cliente	23
Cargue dimensión producto	25
Cargue dimensión fecha	27
Cargue FACT_VENTAS	30
Conclusiones	32
Bibliografía	32

Planteamiento del Problema

Desarrollo de proceso de transformación de datos y carga en el *Data Mart* final.

1. Preparación:
 - a. Revisar el modelo Estrella definido en el proyecto de análisis de la base de datos Jardinería para comprender la estructura y las relaciones entre las tablas de dimensiones y la tabla de hechos.
 - b. Verificar la disponibilidad y consistencia de la base de datos de *staging* previamente creada para el proyecto.
2. Extracción de datos desde la base de datos origen hacia la base de datos de *Staging*:
 - a. Utilizar consultas SQL para extraer datos relevantes de la base de datos origen y cargarlos en las tablas correspondientes de la base de datos de *staging*.
 - b. Verificar la integridad y consistencia de los datos extraídos para asegurar que cumplan con los requisitos del modelo Estrella.
3. Transformación de datos según las necesidades analíticas:
 - a. Aplicar técnicas de transformación de datos, como limpieza, normalización y enriquecimiento, para preparar los datos de acuerdo con las necesidades analíticas específicas.
 - b. Realizar la transformación de los datos utilizando consultas SQL u herramientas de ETL (*Extract, Transform, Load*) según sea necesario para garantizar la calidad y coherencia de los datos.
4. Carga de registros en el *Data Mart* final:
 - a. Diseñar consultas SQL o scripts de carga para insertar los registros transformados desde la base de datos de *staging* en las tablas del *data mart* final.
 - b. Ejecutar las consultas de carga y verificar que los datos se hayan insertado correctamente en el *data mart* final.

5. Documentación y presentación:

- a. Documentar todo el proceso de transformación de datos y carga en un informe detallado que incluya una descripción de las etapas realizadas, las consultas SQL utilizadas.
- b. Presentar el informe de manera clara y concisa, asegurándose de incluir referencias al modelo Estrella y las técnicas de ETL aplicadas.

Introducción

El proceso de Extracción, Transformación y Carga (ETL) es un componente esencial en la construcción de soluciones de inteligencia de negocios. Este informe documenta el desarrollo de un proceso ETL utilizando SQL Server Integration Services (SSIS) para poblar un Data Mart de jardinería, compuesto por varias tablas de dimensiones (cliente, producto, oficina, empleado, pedido, pago y tiempo) y una tabla de hechos de ventas. A través de este proceso, se busca centralizar la información relevante de la empresa para facilitar su análisis y toma de decisiones estratégicas.

Objetivos

1. Diseñar e implementar las tablas de dimensiones y la tabla de hechos siguiendo un modelo estrella.
2. Configurar paquetes SSIS que permitan la carga de datos desde tablas de staging hacia el Data Mart.
3. Asegurar la integridad referencial entre las tablas mediante el uso de claves foráneas.
4. Identificar y resolver errores comunes en la carga, como conflictos de llaves foráneas y metadatos.

Análisis de la base de datos (Problema)

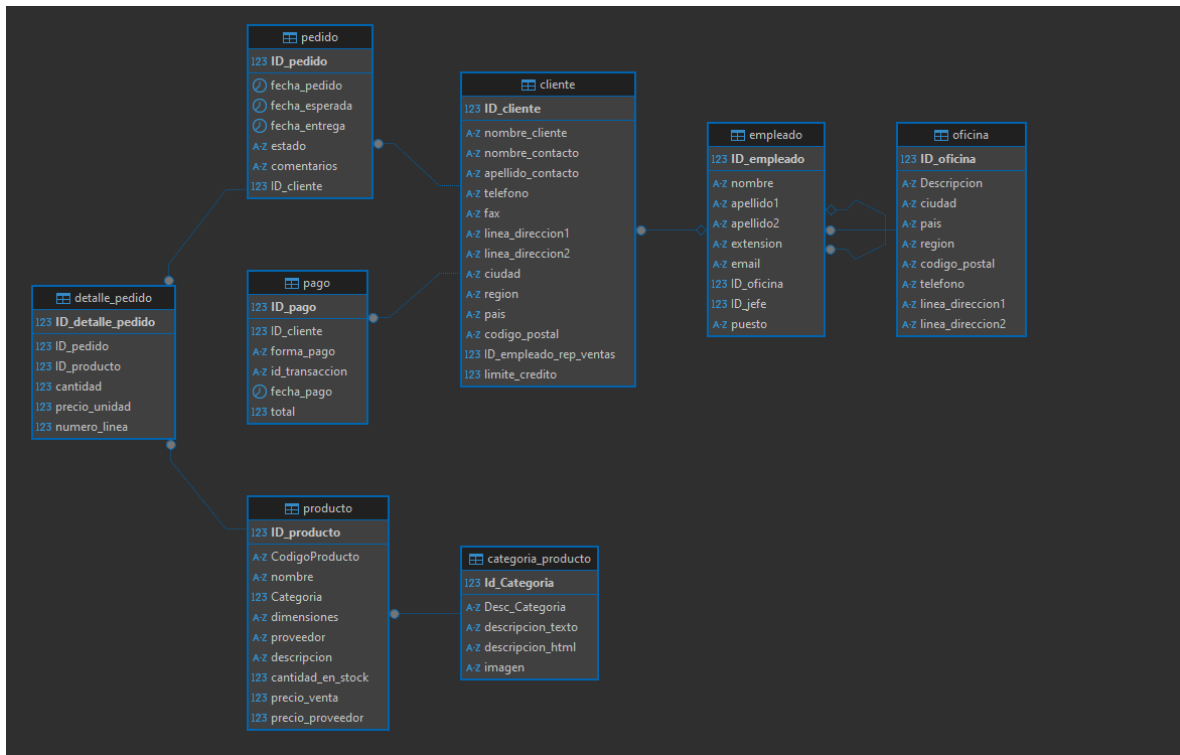


Ilustración 1 Modelo Transaccional Jardinería

1. Se ha cargado la base de datos en la instancia MYSQL y se insertan los datos, así mismo el código cuenta con relaciones en llaves foranes preestablecidas. El producto de esto se muestra en la presente imagen (Ilustración 1)

Identificación de campos necesarios

1. Observamos que la tabla “**detalle_pedido**” es la tabla en la que se concentra el valor transaccional a nivel observable en el esquema de la base de datos de jardinería. En ella encontramos, el detalle del pedido, el identificador del pedido, el identificador del producto y la cantidad.
2. Así mismo observamos que las tablas cliente, producto, empleado, oficina y pago son tablas importantes en este esquema por tanto procederemos a tomarlas en cuenta a la hora de construir el diagrama dimensional.

Diseño del modelo estrella

1. Para comenzar a construir la base de datos Data Mart, necesitaremos primero preparar la base de datos de recepción de datos originarios desde el staging, por tanto, usaremos el siguiente código SQL para crear la base de datos

```
DROP DATABASE IF EXISTS DBO_DATAMART_Jardineria;  
CREATE DATABASE DBO_DATAMART_Jardineria;  
GO  
USE DBO_DATAMART_Jardineria;
```

2. Teniendo lista la base de datos podremos desde el SISS realizar la transformación y traslado de la información desde la Base de datos Staggin el cual ya insertamos en la evidencia pasada.
3. Retomando a su vez el desarrollo del modelo estrella que hicimos en la actividad 1, retomamos que las dimensiones serán las siguientes, producto,

empleado, oficina y pago que tomarán los nombres de:

- a. **dim_tiempo** (Basada en fechas de pedido y pago)
- b. **dim_cliente** (Basada en la tabla cliente)
- c. **dim_producto** (Basada en la tabla producto)
- d. **dim_empleado** (Basada en la tabla empleado)
- e. **dim_oficina** (Ubicación de empleados)
- f. **dim_pago** (Basada en las transacciones de pago)

EXTRAS:

- g. **dim_pedido** (Basada en el estado de pedido y método de envío)

4. Las relaciones entre la tabla de hechos y las dimensiones serán de la siguiente manera:

h. FK > ID_CLIENTE	dim_cliente	-> ID_CLIENTE
i. FK > ID_PRODUCTO	dim_producto	-> ID_PRODUCTO
j. FK > ID_EMPLEADO	dim_empleado	-> ID_EMPLEADO
k. FK > ID_OFICINA	dim_oficina	-> ID_OFICINA
l. FK > ID_TIEMPO	dim_tiempo	-> ID_TIEMPO
m. FK > ID_PAGO	dim_pago	-> ID_PAGO
EXTRAS_		
n. FK > ID_PEDIDO	dim_pedido	-> ID_PEDIDO

Representación Gráfica del modelo estrella

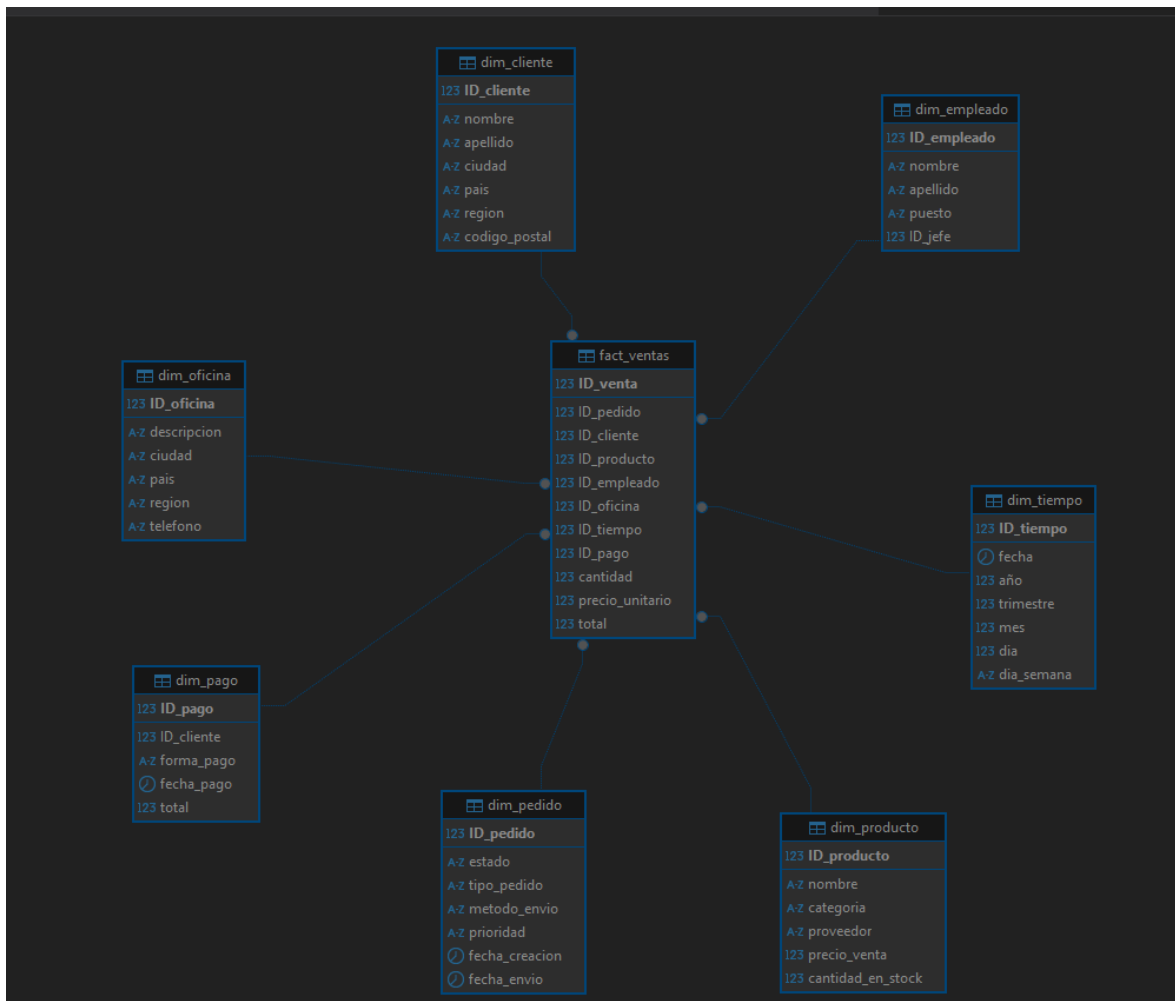
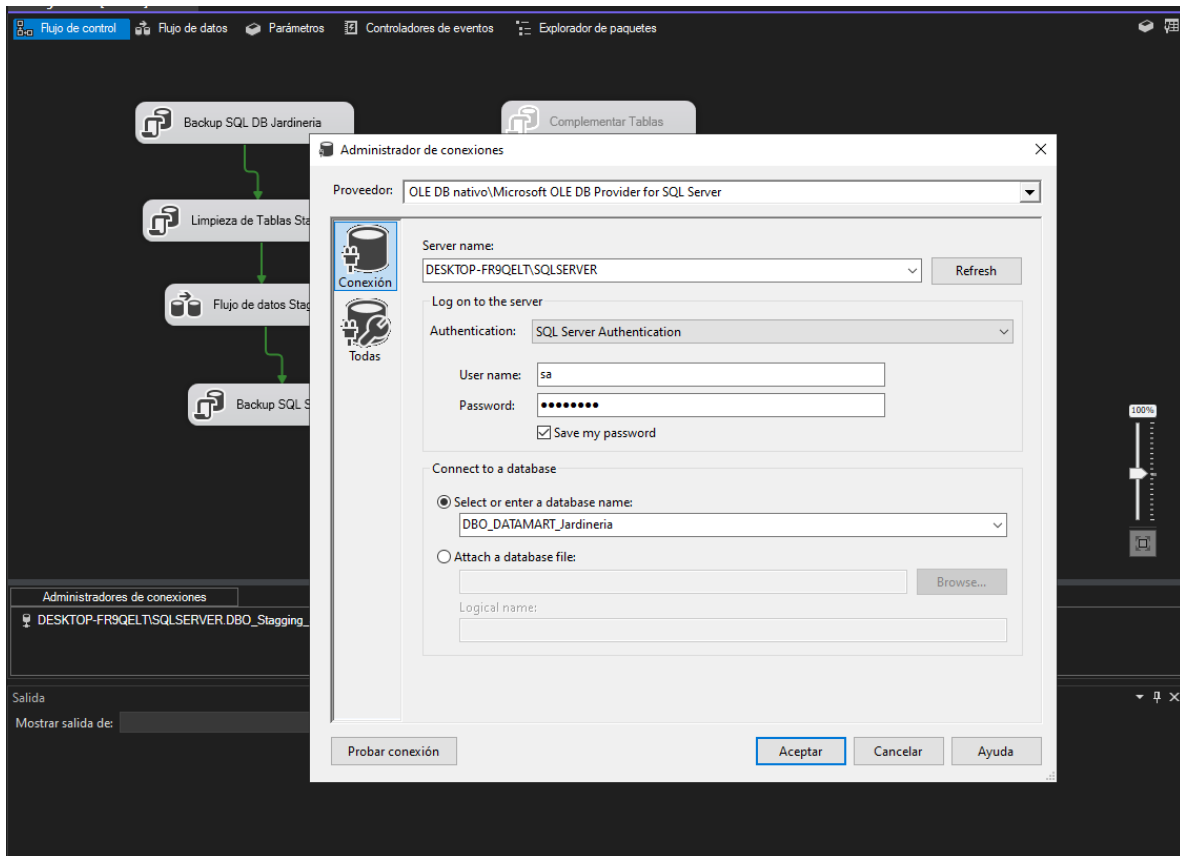


Ilustración 2 Modelo Dimensional

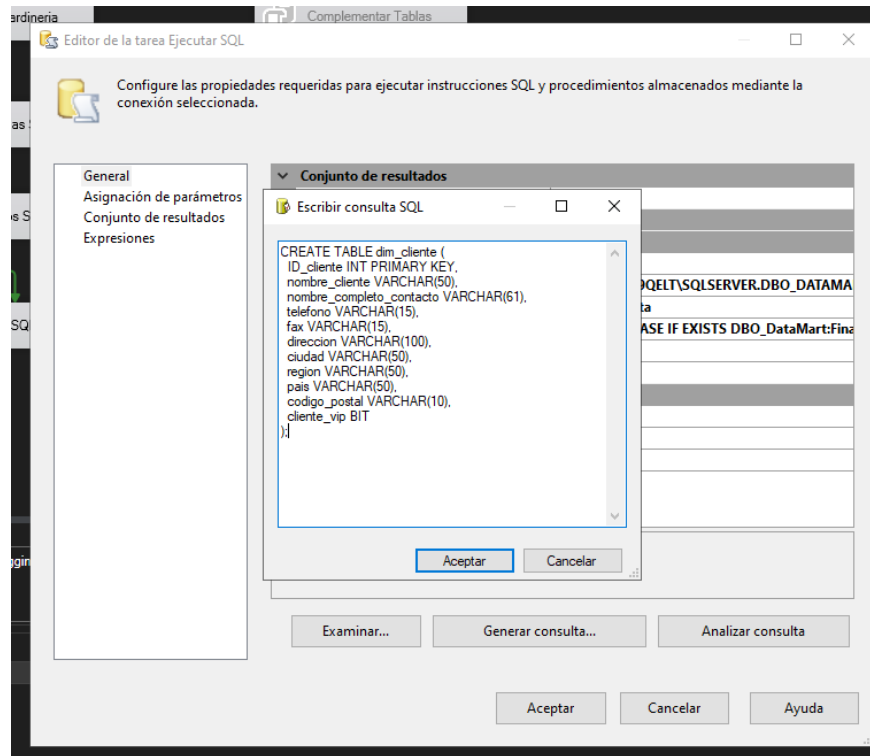
Construcción de tablas desde herramienta SSIS



Teniendo la base de datos ya creada por parte del área de base de datos podremos comenzar a programar las tareas de traslado, pero primero necesitaremos construir las tablas que recibirán la información.

Crear tabla de dimensiones

Tabla dimensión cliente

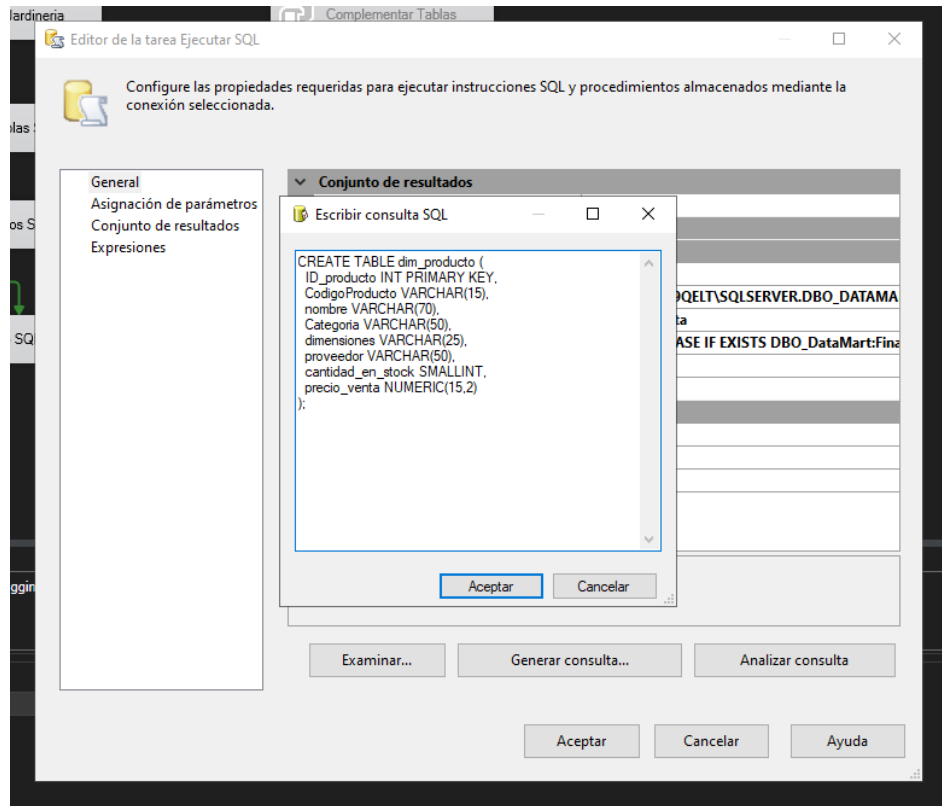


```
1 CREATE TABLE dim_cliente (  
2   ID_cliente INT PRIMARY KEY,  
3   nombre_cliente VARCHAR(50),  
4   nombre_completo_contacto VARCHAR(61),  
5   telefono VARCHAR(15),  
6   fax VARCHAR(15),  
7   direccion VARCHAR(100),  
8   ciudad VARCHAR(50),  
9   region VARCHAR(50),  
10  pais VARCHAR(50),  
11  codigo_postal VARCHAR(10),  
12  cliente_vip BIT  
13 );
```

Transformaciones a realizar

- nombre_completo_contacto = nombre_contacto + ' ' + apellido_contacto
- direccion = concat(linea_direccion1, ' ', linea_direccion2)
- cliente_vip = 1 si limite_credito > 100000

Tabla Dimensión producto

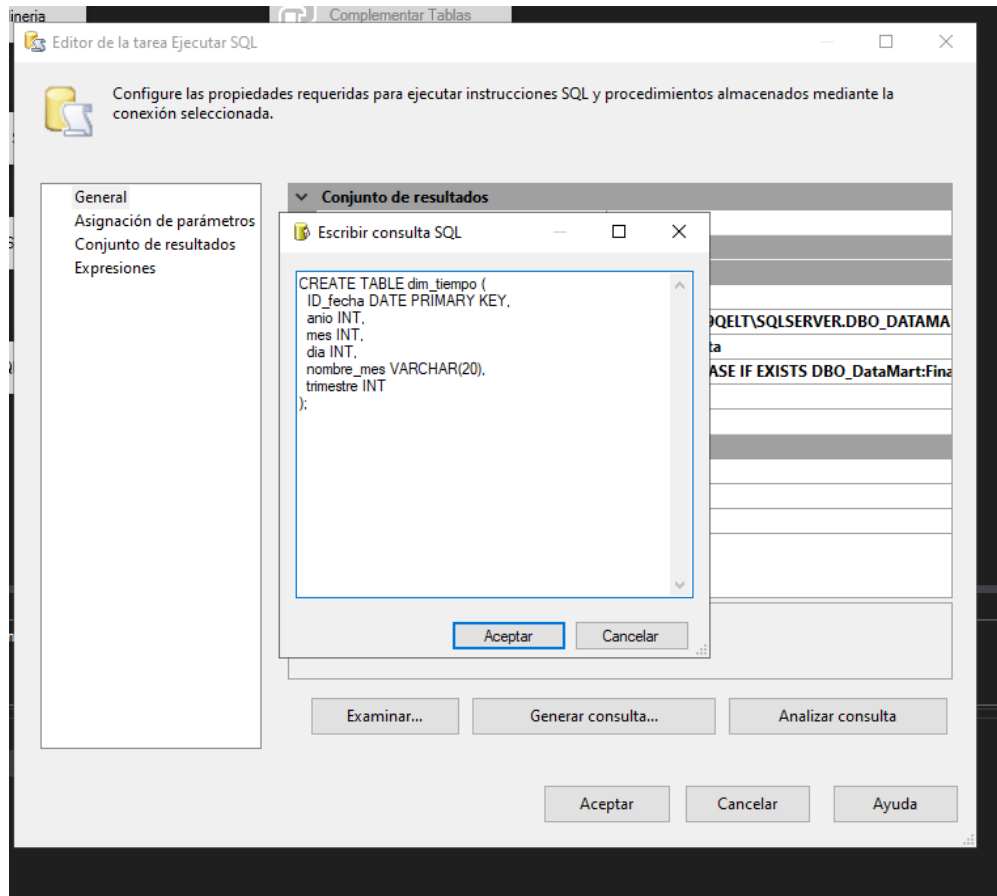


```
1 CREATE TABLE dim_producto (
2   ID_producto INT PRIMARY KEY,
3   CodigoProducto VARCHAR(15),
4   nombre VARCHAR(70),
5   Categoria VARCHAR(50),
6   dimensiones VARCHAR(25),
7   proveedor VARCHAR(50),
8   cantidad_en_stock SMALLINT,
9   precio_venta NUMERIC(15,2)
10 );
```

Transformaciones a realizar

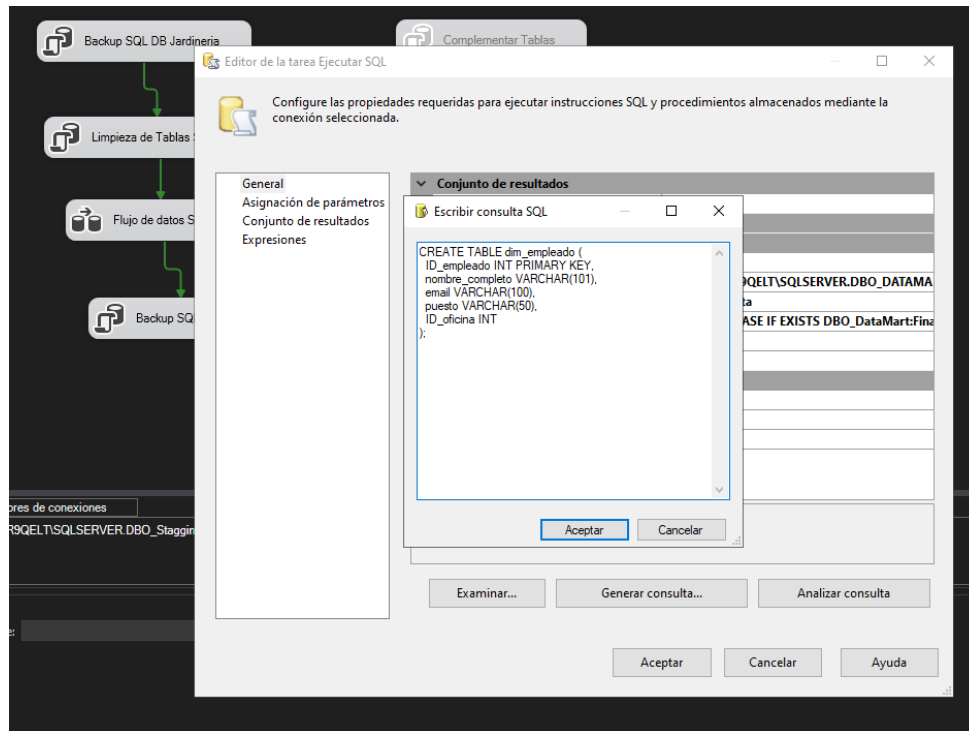
- La categoría viene de la tabla Categoria_producto.

Tabla dimensión tiempo



```
1 CREATE TABLE dim_tiempo (  
2     ID_fecha DATE PRIMARY KEY,  
3     anio INT,  
4     mes INT,  
5     dia INT,  
6     nombre_mes VARCHAR(20),  
7     trimestre INT  
8 );
```

Tabla dimensión empleado

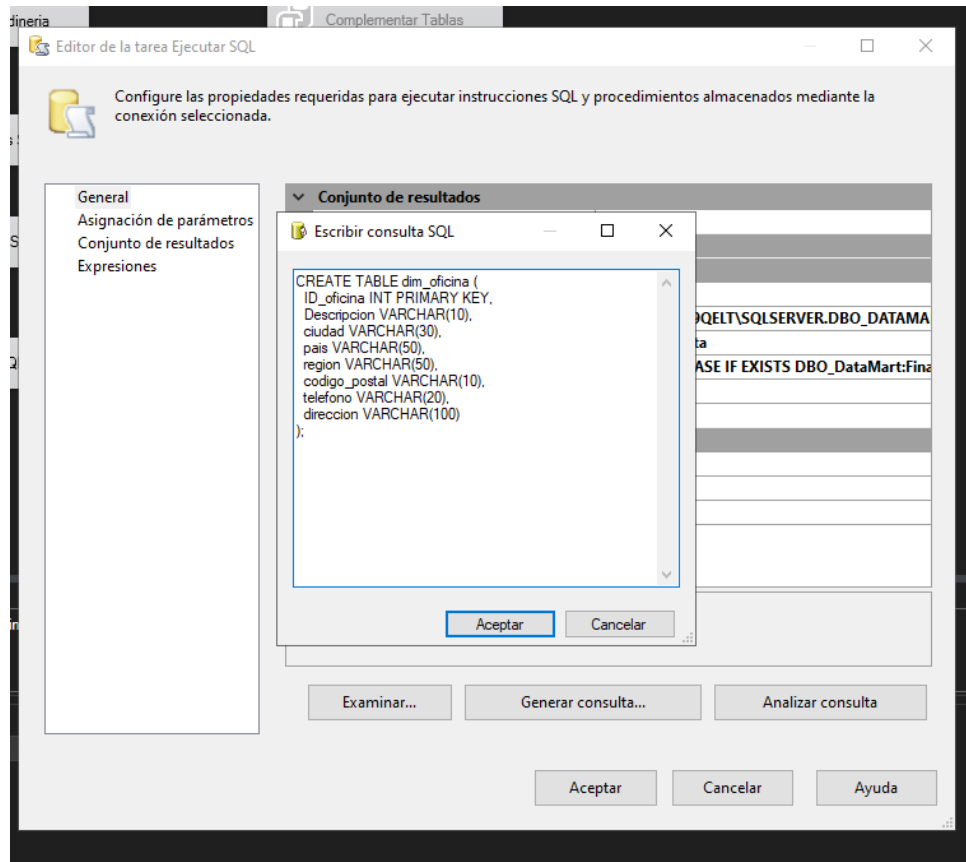


```
1 CREATE TABLE dim_empleado (  
2   ID_empleado INT PRIMARY KEY,  
3   nombre_completo VARCHAR(101),  
4   email VARCHAR(100),  
5   puesto VARCHAR(50),  
6   ID_oficina INT  
7 );
```

Transformaciones a realizar

- Concatenar el nombre_completo = nombre + ' ' + apellido1 + ' ' + apellido2

Tabla dimensión oficina

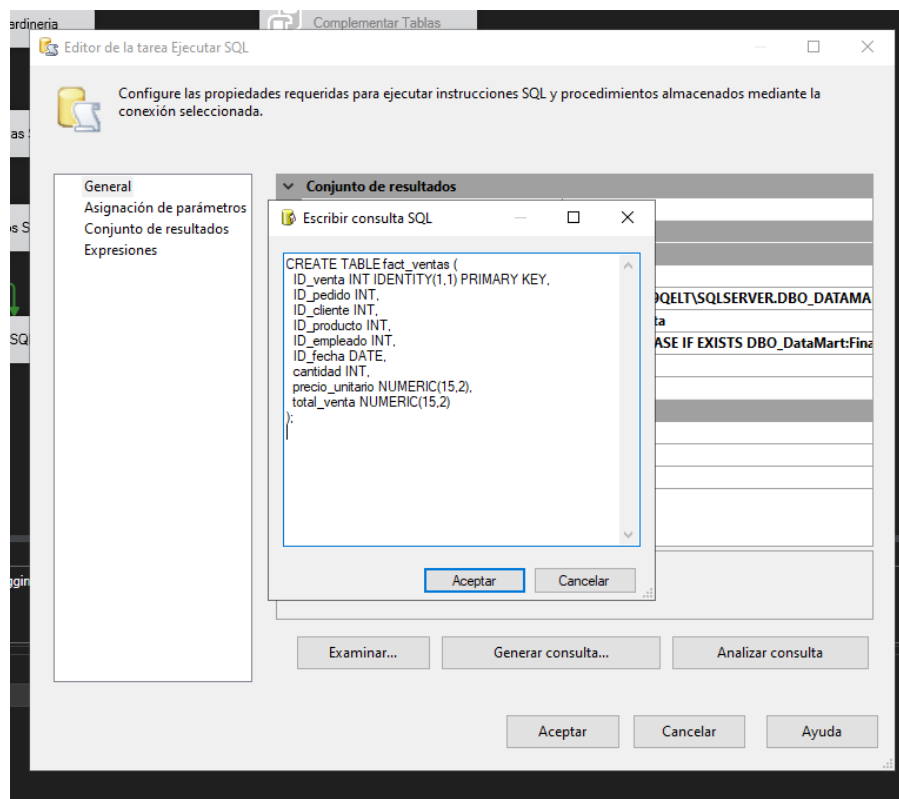


```
1 CREATE TABLE dim_oficina (
2   ID_oficina INT PRIMARY KEY,
3   Descripcion VARCHAR(10),
4   ciudad VARCHAR(30),
5   pais VARCHAR(50),
6   region VARCHAR(50),
7   codigo_postal VARCHAR(10),
8   telefono VARCHAR(20),
9   direccion VARCHAR(100)
10 );
```

Transformaciones a realizar

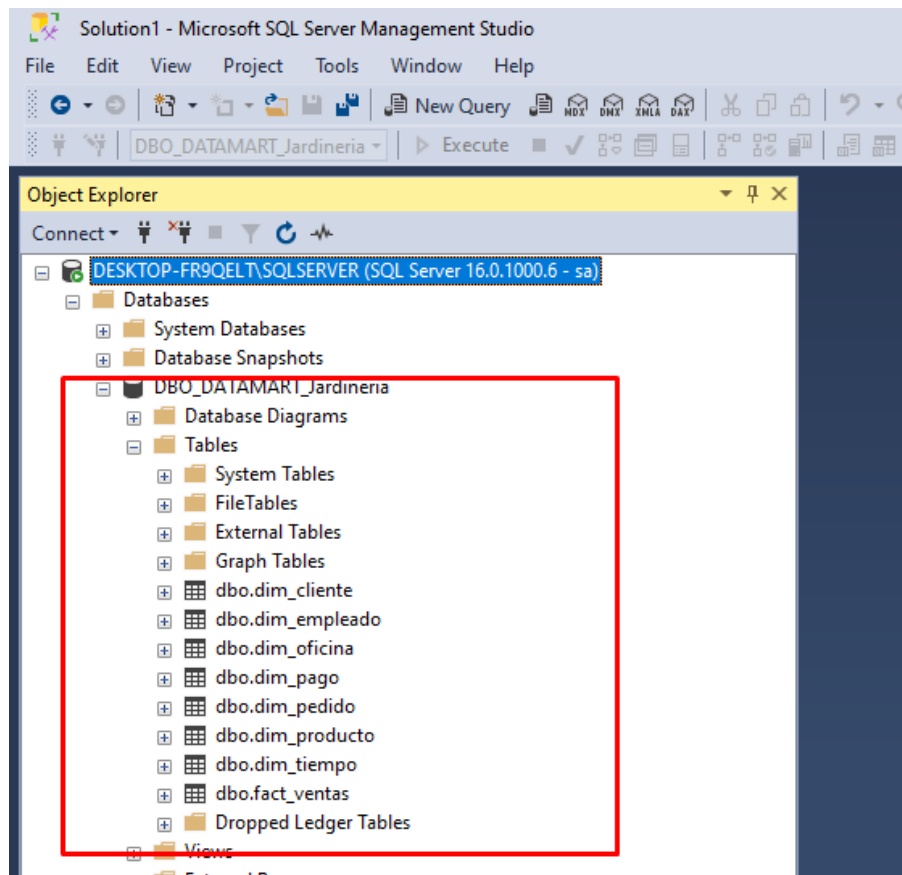
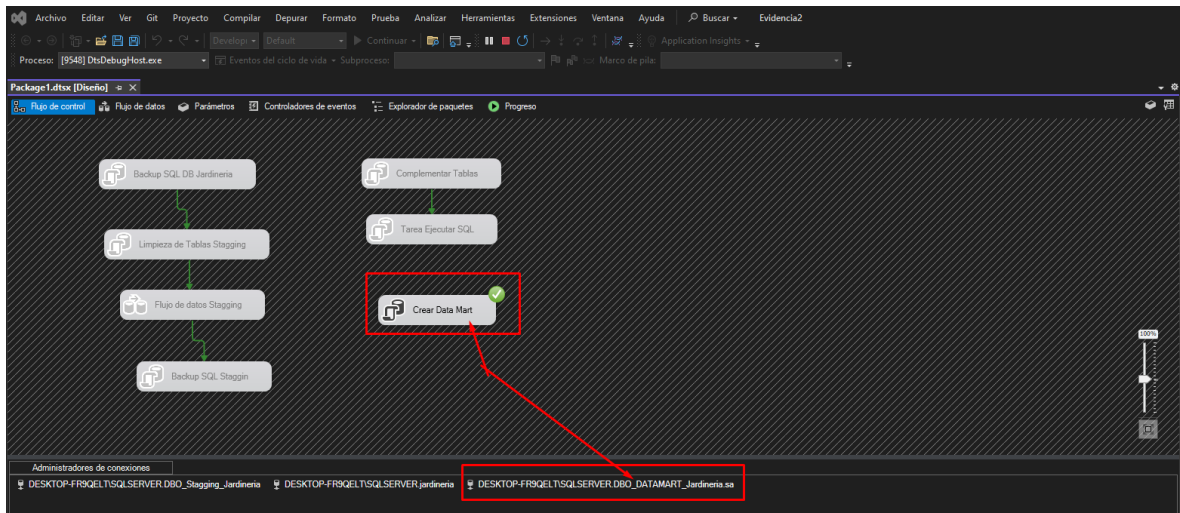
- Conversión de dirección = concat(linea_direccion1, ' ', linea_direccion2)

Tabla de hechos ventas



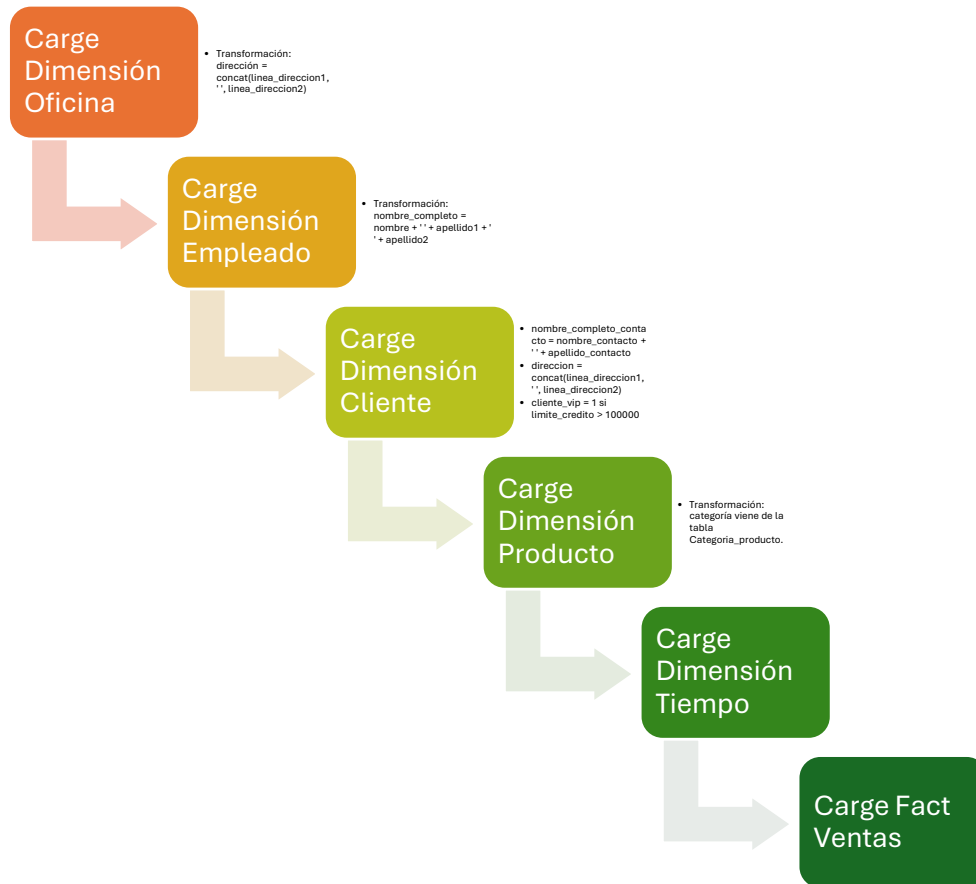
```
1 CREATE TABLE fact_ventas (
2   ID_venta INT IDENTITY(1,1) PRIMARY KEY,
3   ID_pedido INT,
4   ID_cliente INT,
5   ID_producto INT,
6   ID_empleado INT,
7   ID_fecha DATE,
8   cantidad INT,
9   precio_unitario NUMERIC(15,2),
10  total_venta NUMERIC(15,2)
11 );
```

Ejecutamos la tarea y validamos los resultados

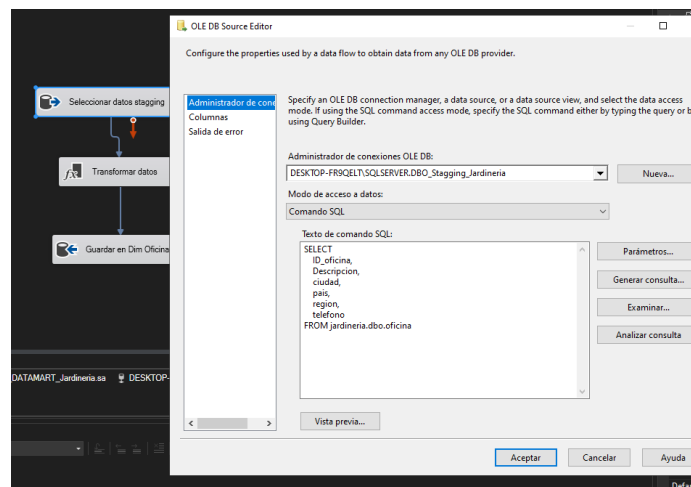
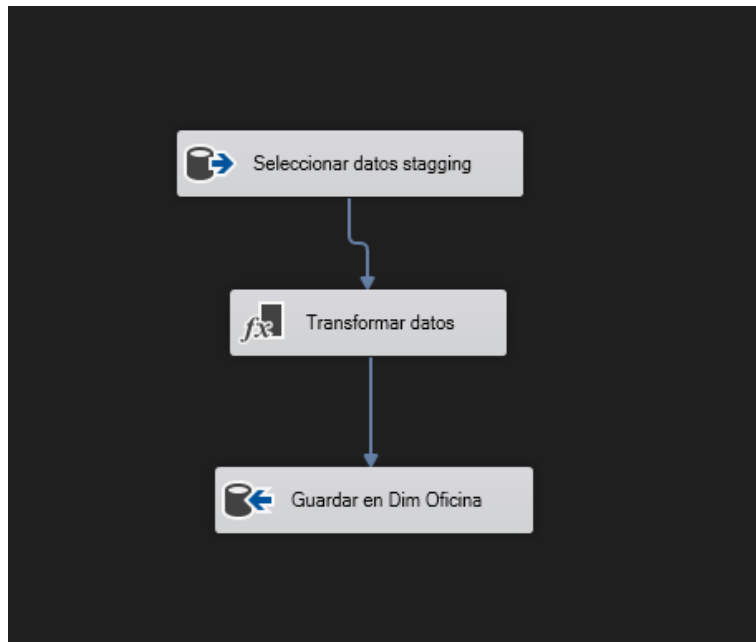


Con esto ya tenemos todo preparado para realizar la transformación y envío de datos al Data Mart

Flujo de Data Flow Task



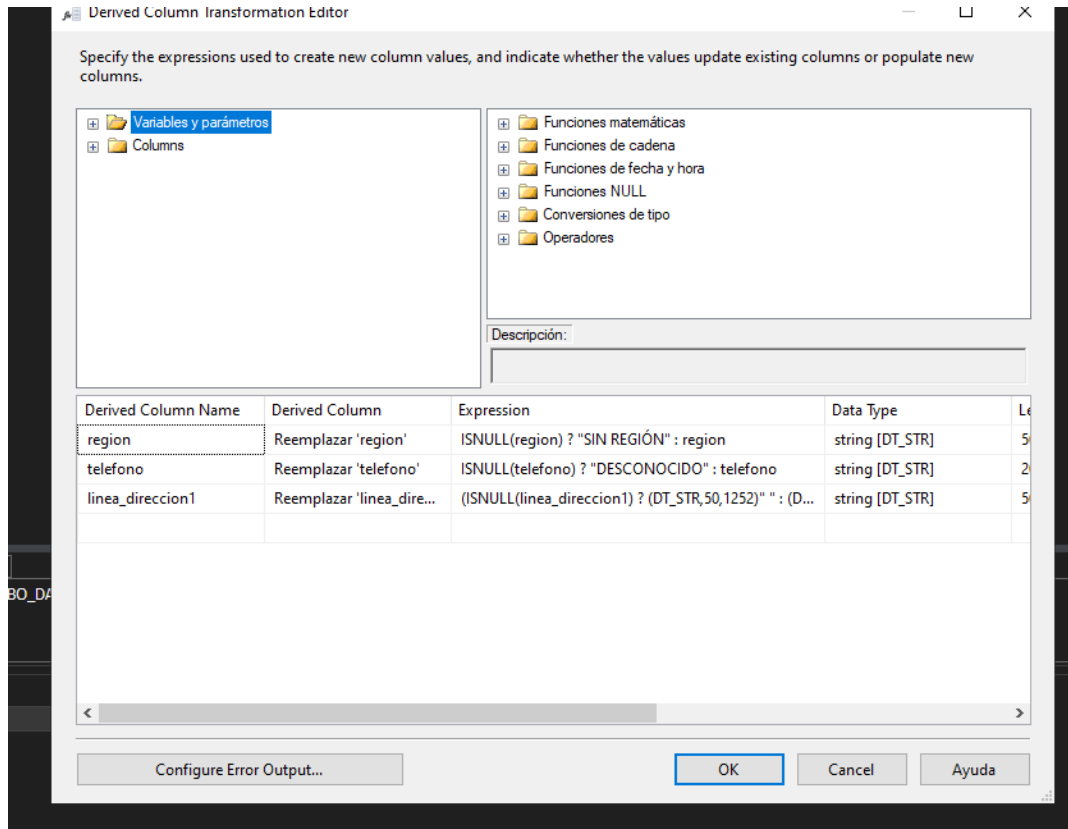
Cargue de Dimensión oficina



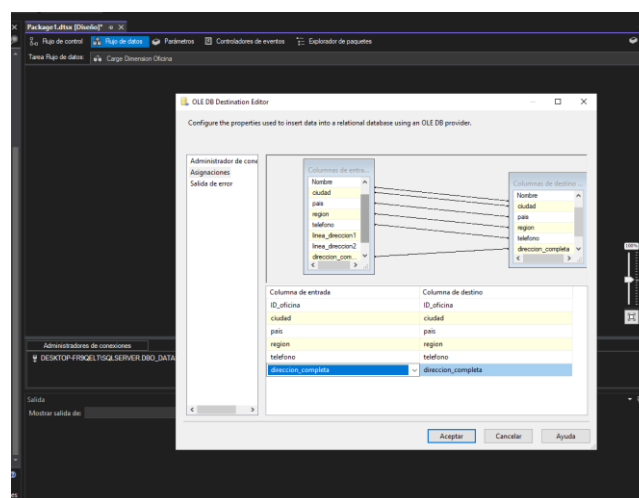
Para la tabla de oficina vamos a hacer una selección de los campos previo a la transformación de los datos, usaremos la siguiente consulta

```
1 SELECT
2     ID_oficina,
3     ciudad,
4     país,
5     region,
6     telefono,
7     linea_direccion1 ,
8     linea_direccion2
9 FROM jardineria.dbo.oficina
```

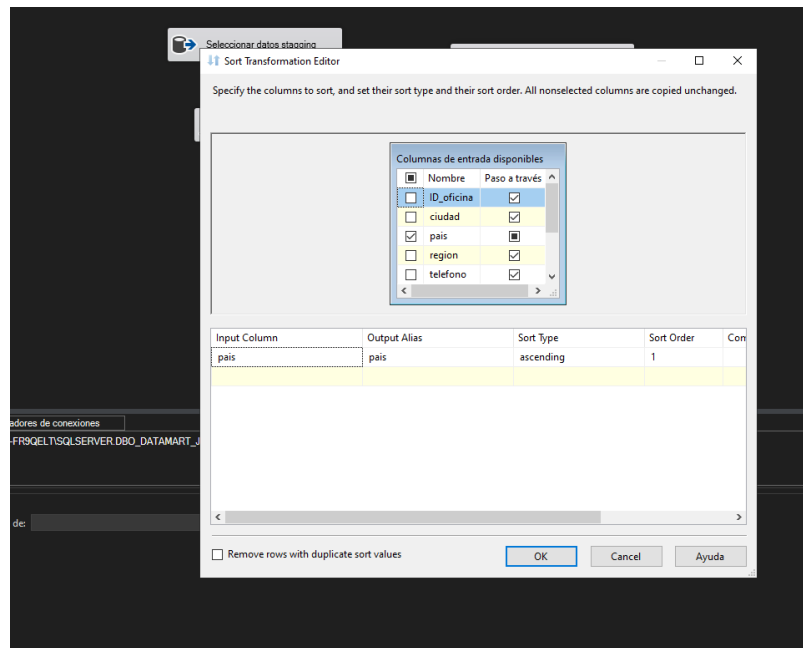
Vamos a concatenar las líneas de direcciones en una sola la conservaremos y remplazaremos la línea dirección 1 y esa se asignara a la dimensión como columna **dimensión_completa**



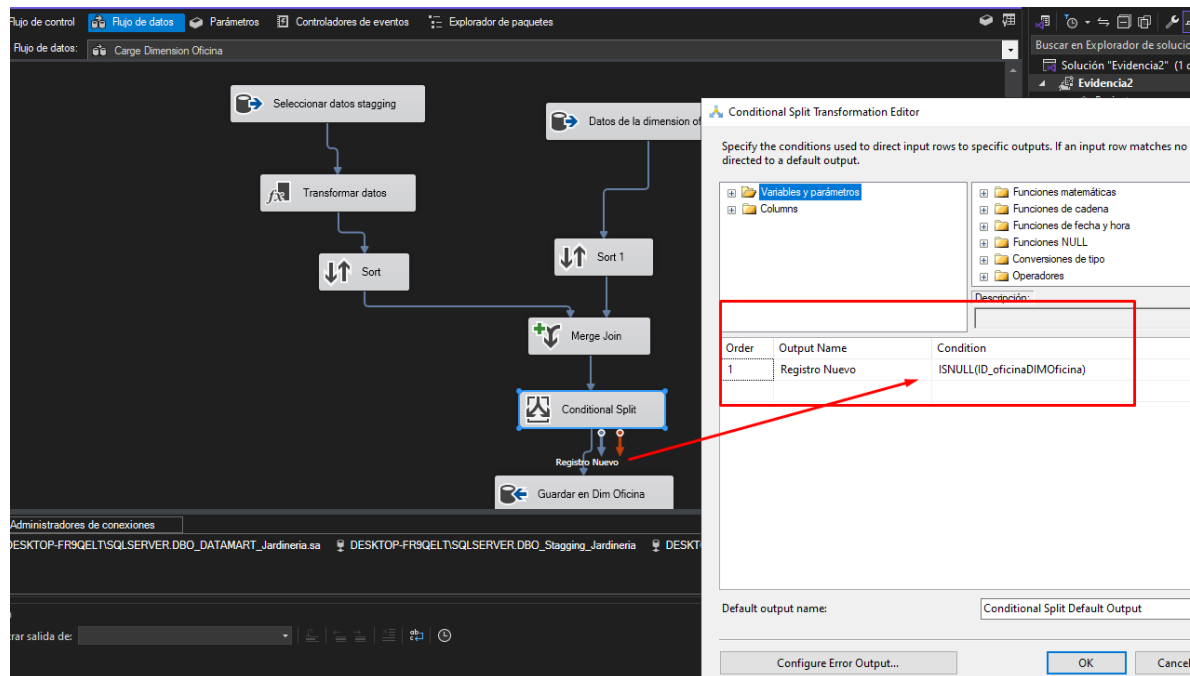
A partir de los datos resultantes obtendremos una sola dirección, valores por defectos en caso de null

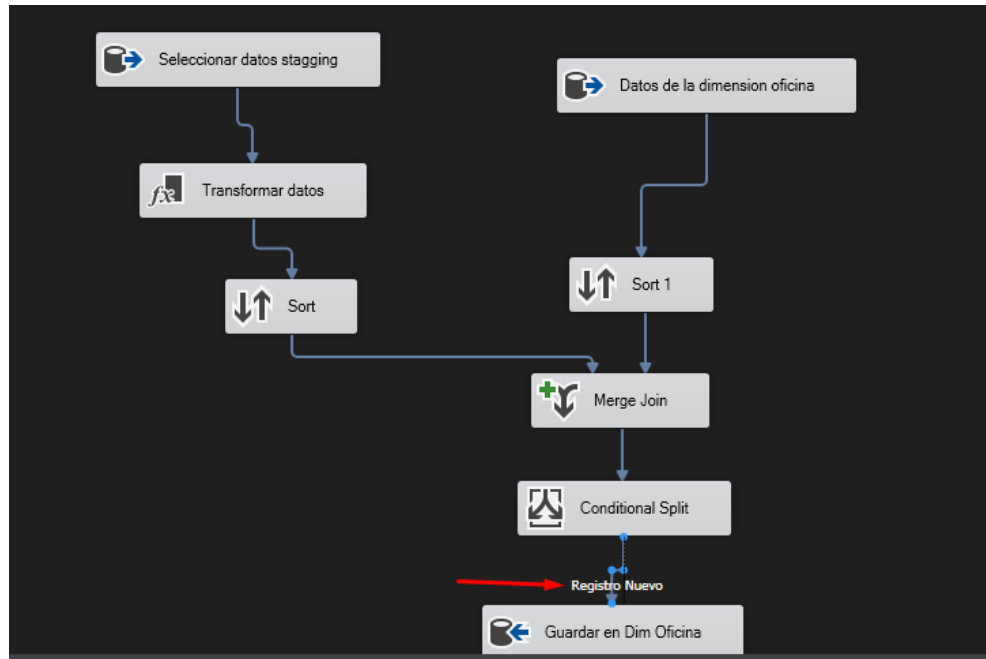


Agregamos un sort para ordenar los datos por país



Adicionalmente debemos verificar si el registro es nuevo, Esto lo haremos por medio del **merge Join** y el condicional Split, uniendo los registros e insertando solo si este último es nuevo





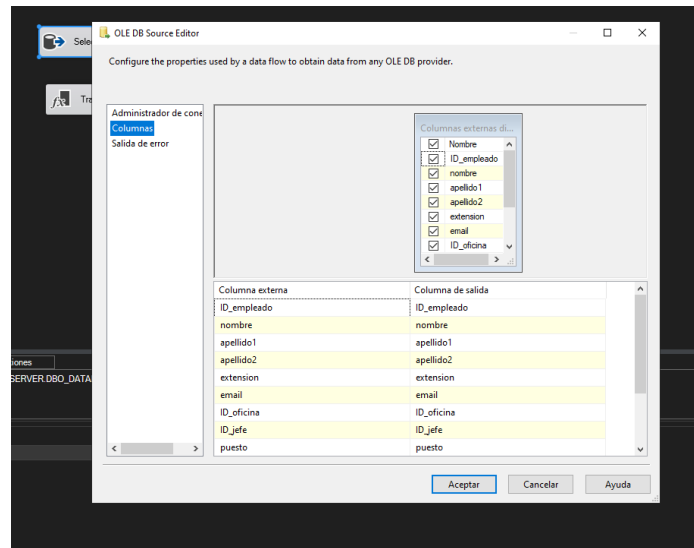
De esta forma tendremos la transformación de la data desde staging a la dimensión Oficina del modelo Estrella de nuestro Data Mart.

Procederemos a repetir el proceso con las demás tablas de dimensión

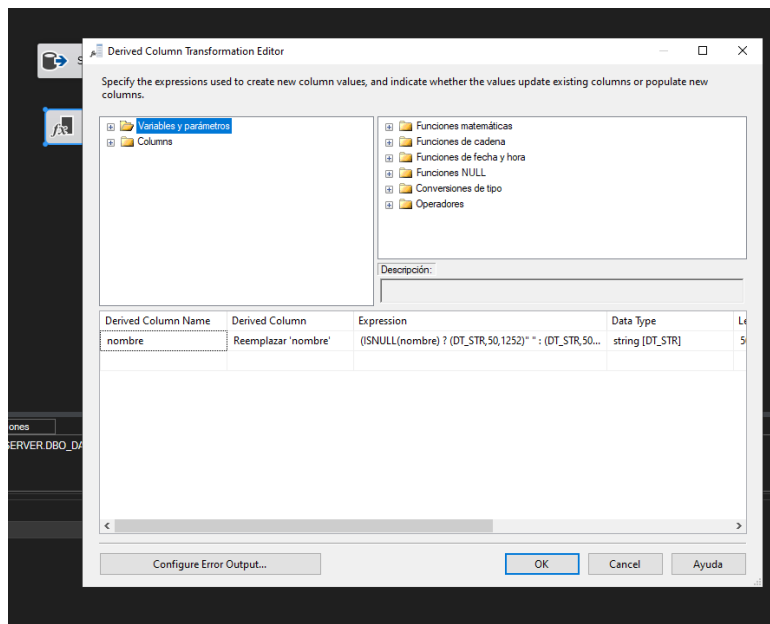
Cargue dimensión empleado

Rápidamente haremos el mismo proceso

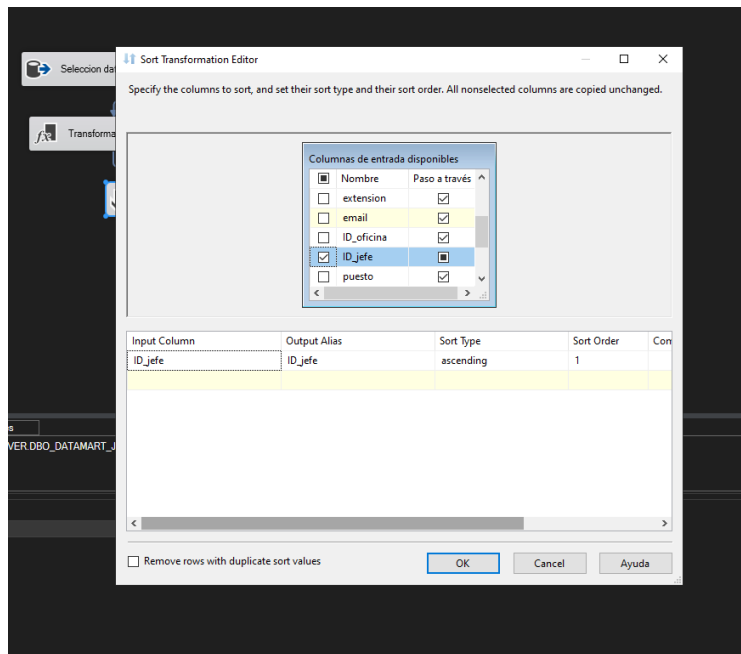
Consultamos todos los datos



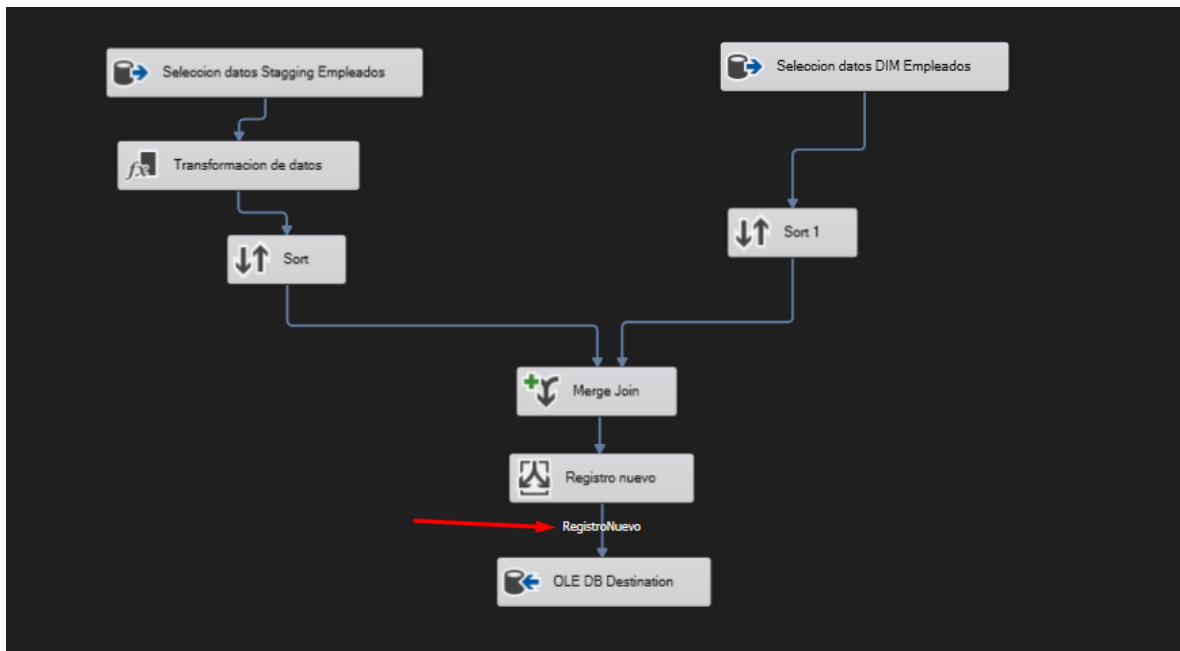
Concatenamos todos los datos del nombre para obtener un nombre completo



Ordenamos por el id del jefe inmediato



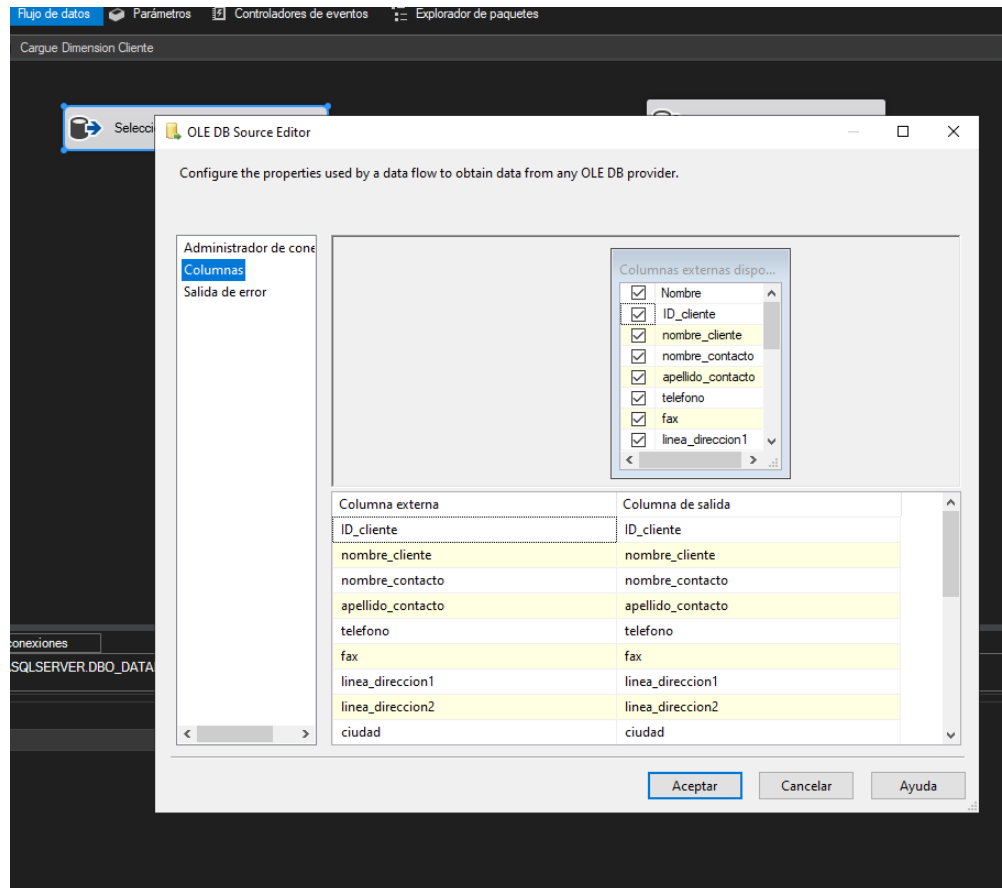
Aplicamos la validación de registro nuevo y la escritura



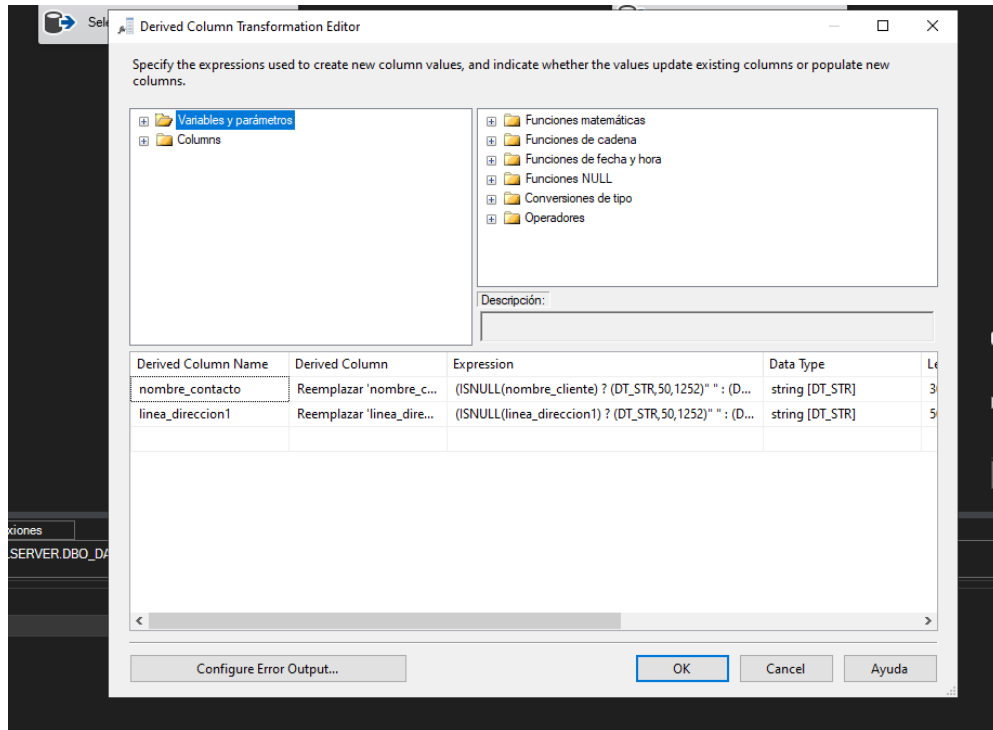
Cargue Dimensión Cliente

Repetimos el proceso como hemos hecho anteriormente

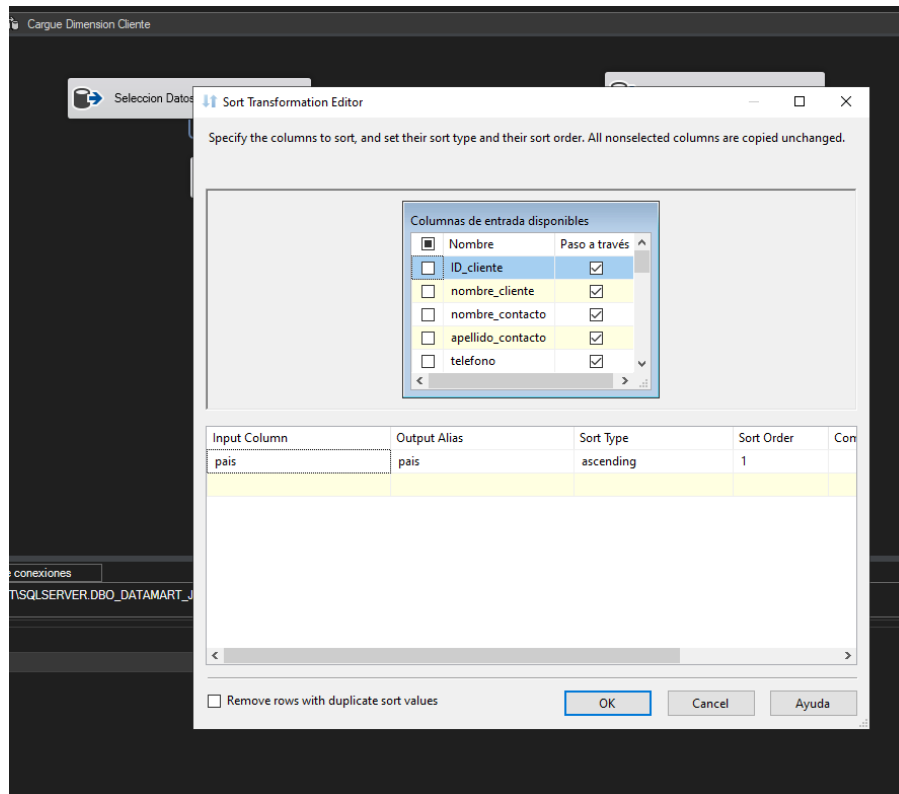
Seleccionamos los datos



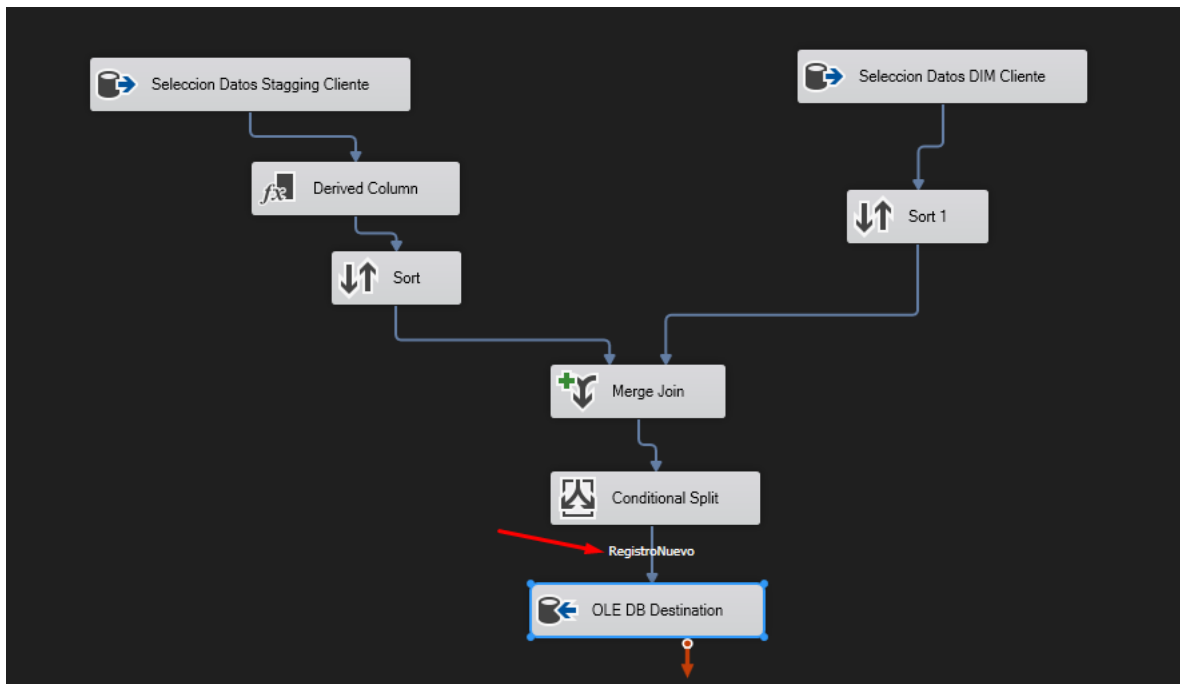
Concatenamos el nombre uniendo apellido y nombre y concatenamos la línea de dirección y los guardamos en los atributos **nombre_contacto** y **línea_direccion1** correspondientemente



Ordenamos por país

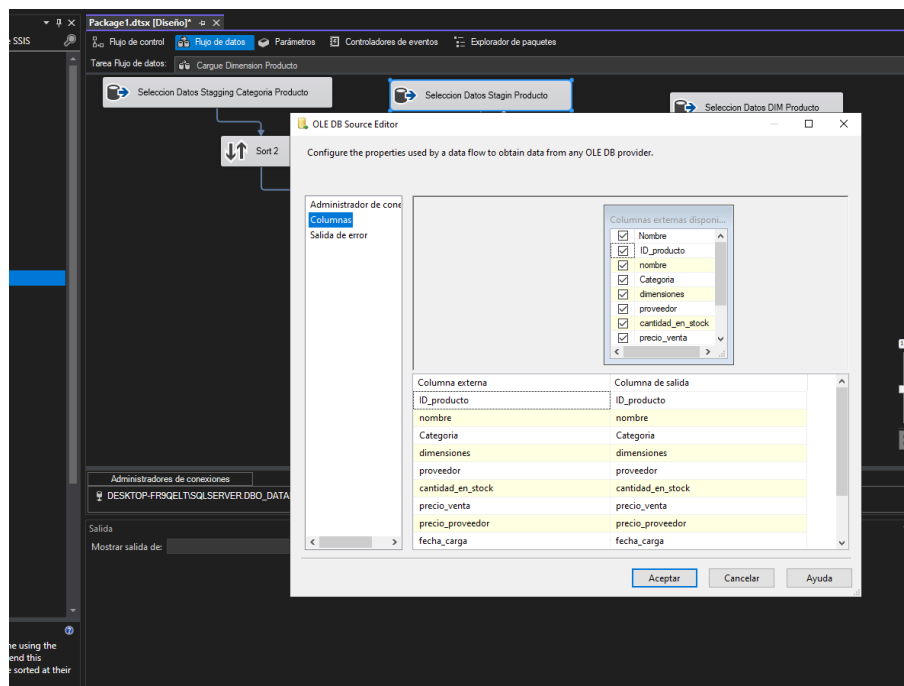


Registramos los registros nuevos

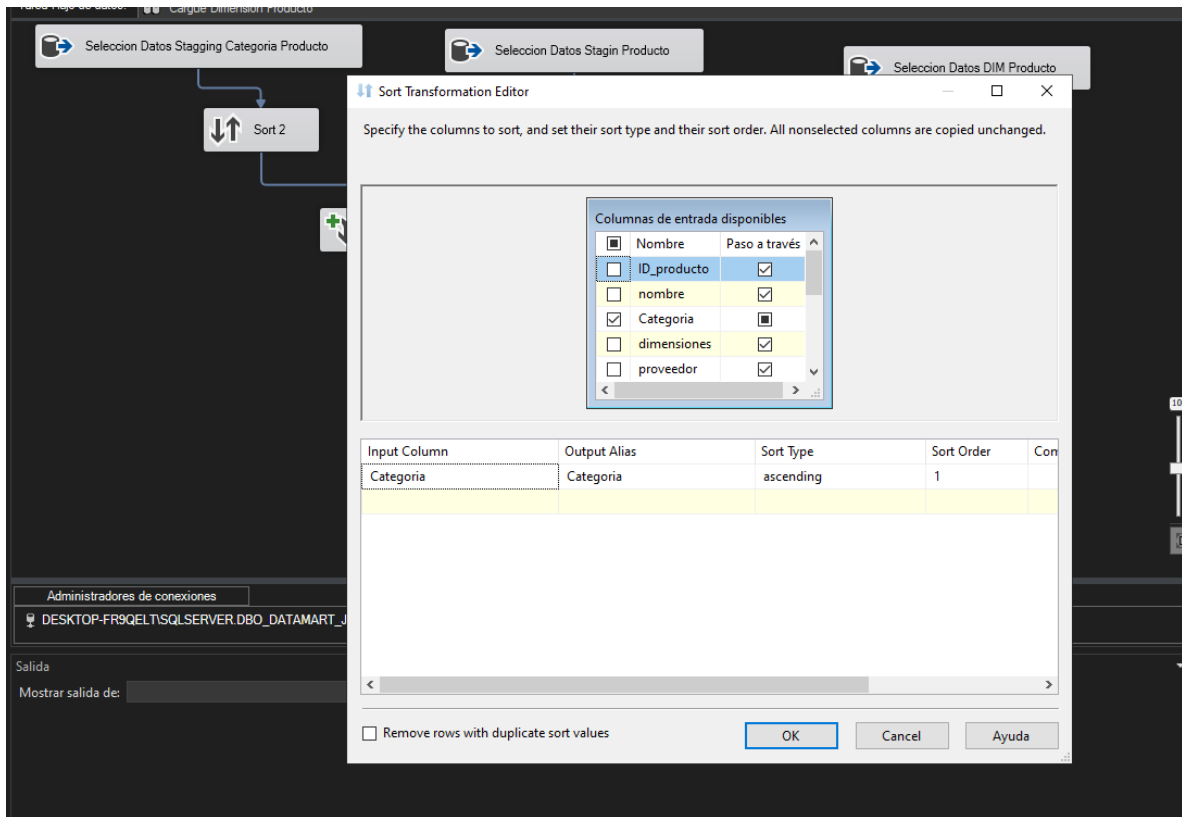


Cargue dimensión producto

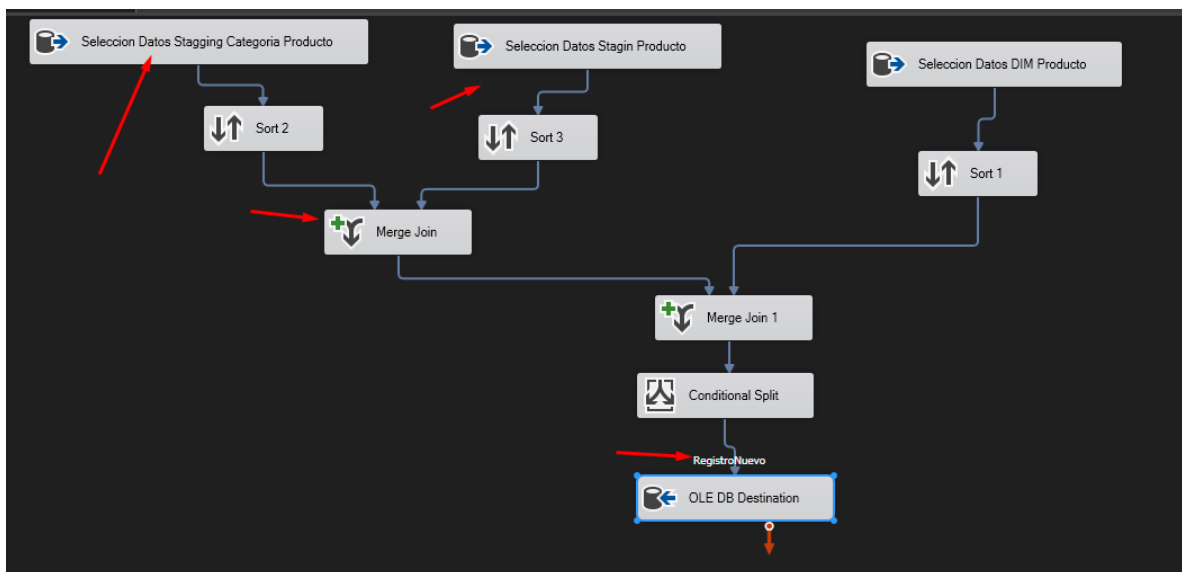
Seleccionamos los datos de categoría producto y producto para hacer merge con la descripción, los datos de categoría tendrán la descripción



Ordenamos por categoría



Filtramos y hacemos un registro de solo los datos nuevos

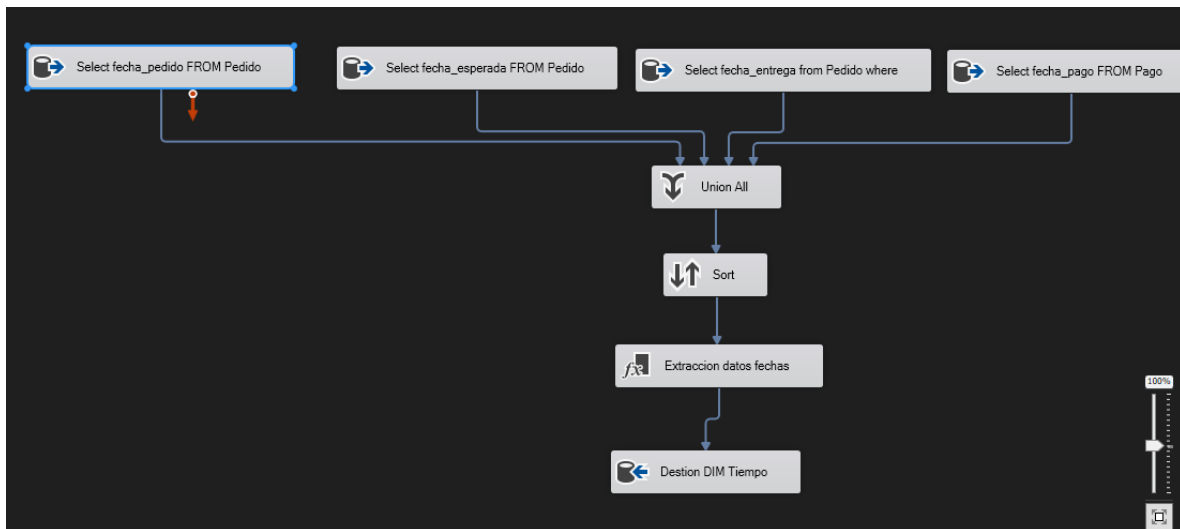


Cargue dimensión fecha

Para la dimensión tiempo hay que ser mas cuidadoso, el proceso buscará obtener las fechas correctas de los pedidos y los pagos para obtener correctamente el registro de eventos.

El flujo será el siguiente

- Uno para SELECT fecha_pedido FROM pedido
- Otro para SELECT fecha_esperada FROM pedido
- Otro para SELECT fecha_entrega FROM pedido WHERE fecha_entrega IS NOT NULL
- Otro para SELECT fecha_pago FROM pago



Codigo SQL

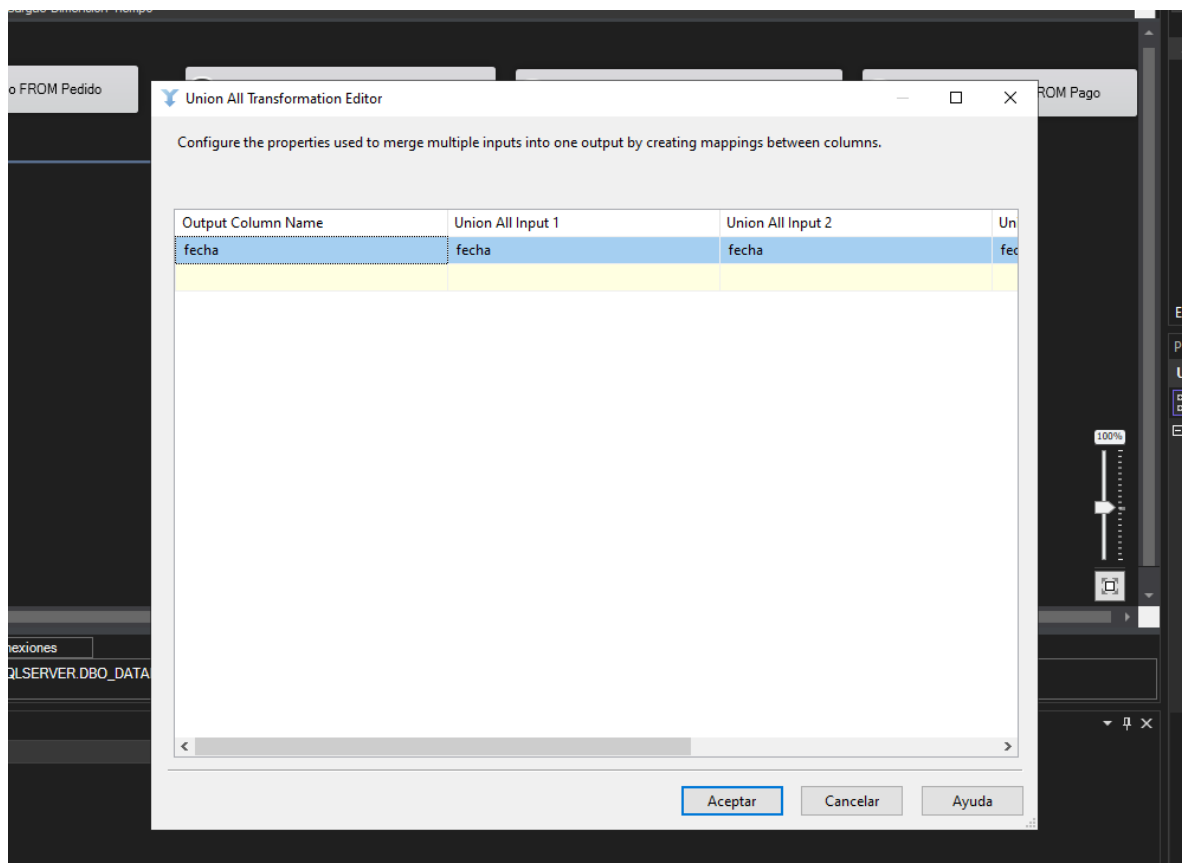
```
1 SELECT DISTINCT fecha_pedido as fecha
2 FROM [DB Staging Pedido]
3 WHERE (fecha_pedido IS NOT NULL)
```

```
1 SELECT DISTINCT fecha_esperada as fecha
2 FROM [DB Staging Pedido]
3 WHERE (fecha_esperada IS NOT NULL)
```

```
1 SELECT DISTINCT fecha_entrega as fecha
2 FROM [DB Stagging Pedido]
3 WHERE (fecha_entrega IS NOT NULL)
```

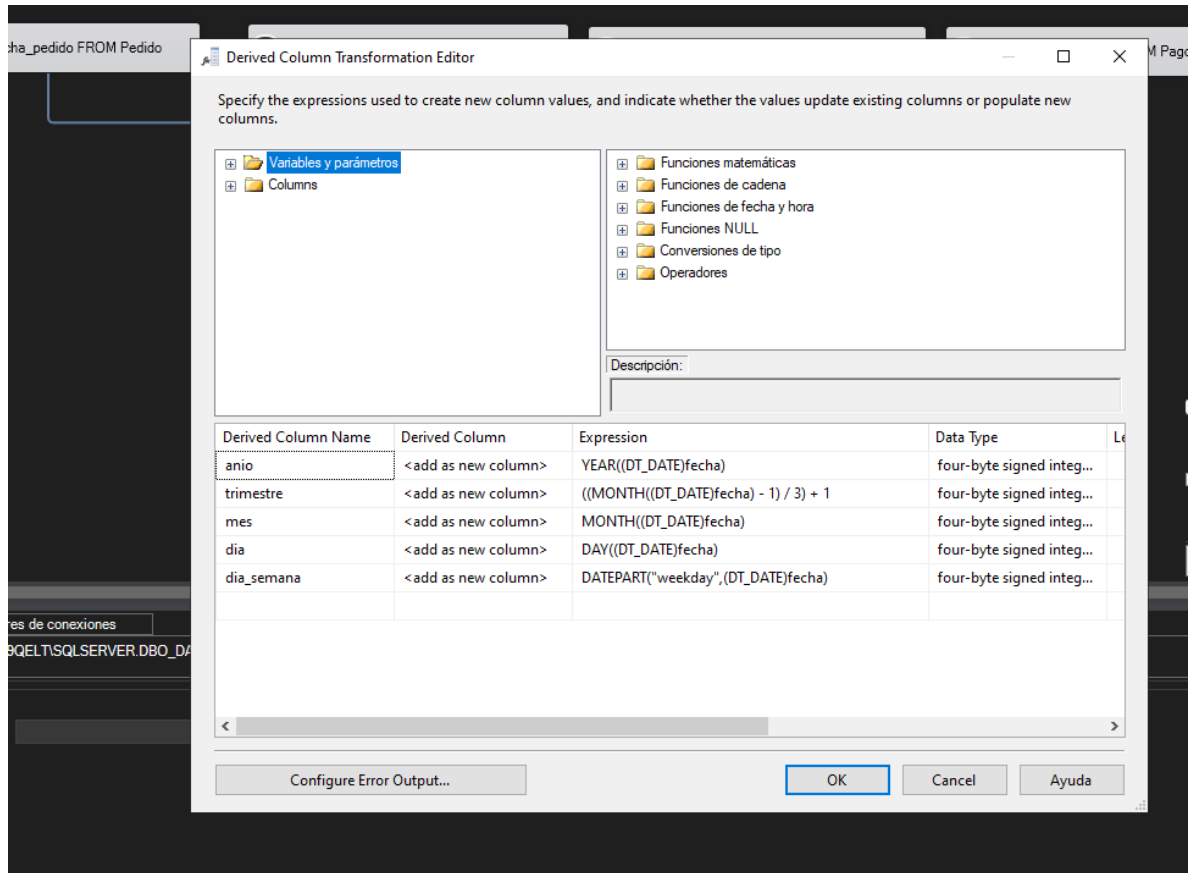
```
1 SELECT DISTINCT fecha_pago AS fecha
2 FROM [DB Stagging Pedido]
3 WHERE fecha_pago IS NOT NULL
```

Hacemos las correspondientes uniones

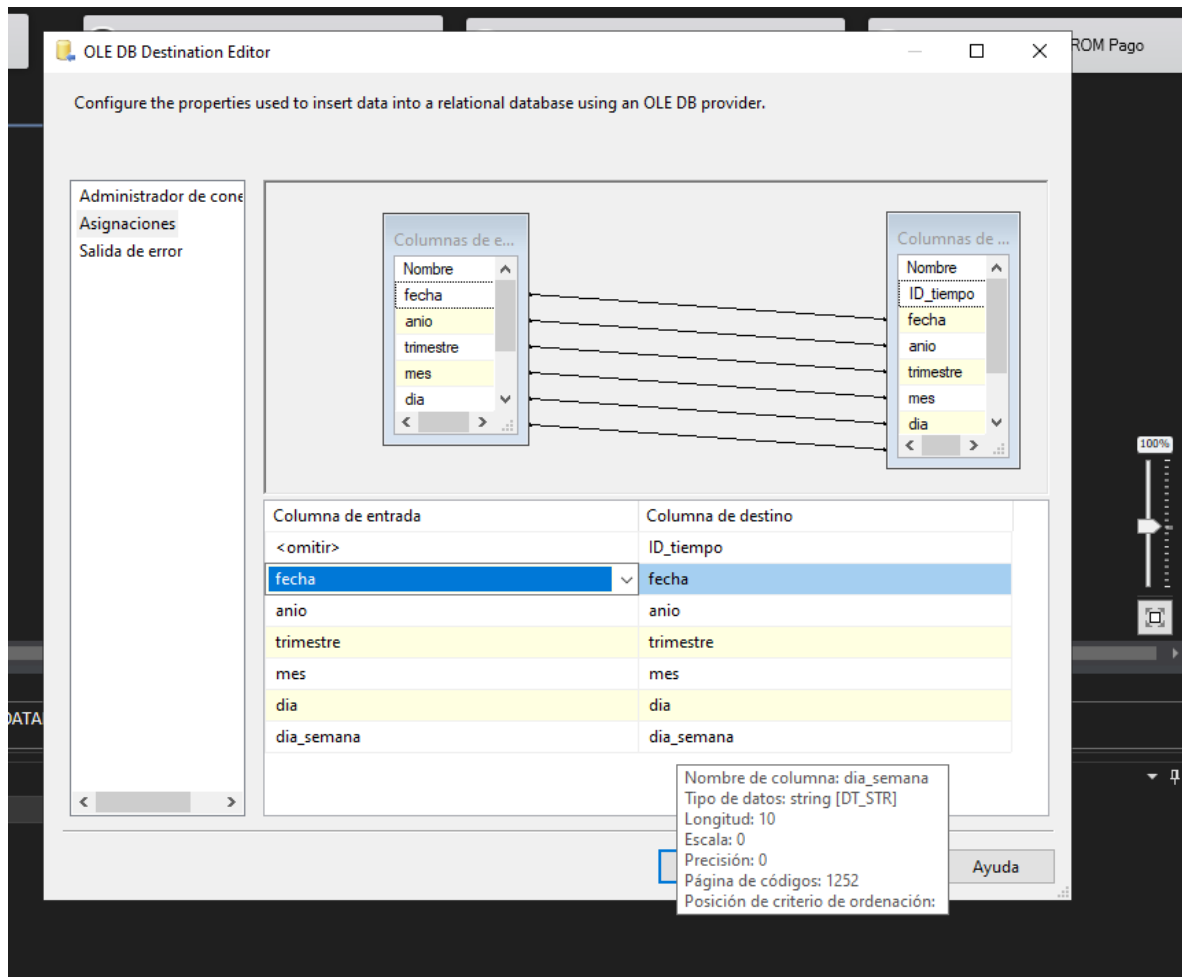


Hacemos la transformación de la data

```
1 YEAR((DT_DATE) fecha)
2 ((MONTH((DT_DATE) fecha) - 1) / 3) + 1
3 MONTH((DT_DATE) fecha)
4 DAY((DT_DATE) fecha)
5 DATEPART("weekday", (DT_DATE) fecha)
```

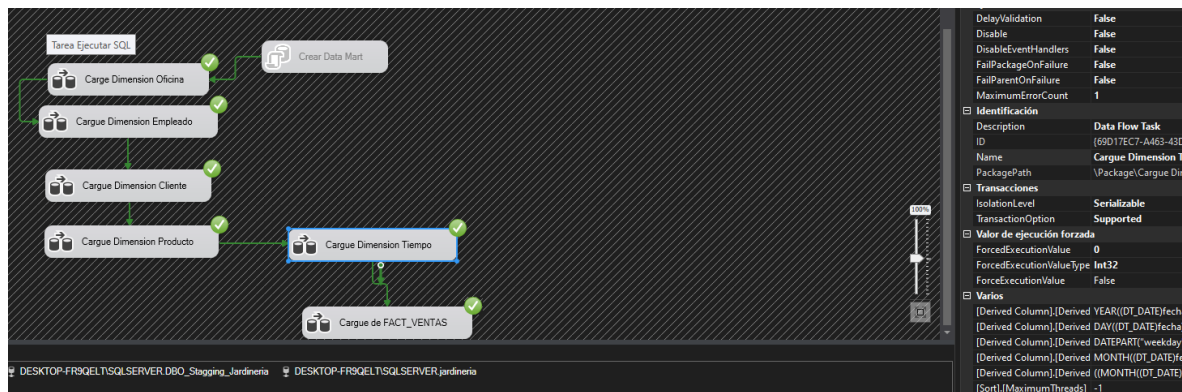


Finalmente hacemos la relación de los datos de las fechas



Cargue FACT_VENTAS

Finalmente validamos el cargue de los datos y su registro en la base de datos



```

1 SELECT TOP (1000) [ID_cliente]
2     , [nombre_cliente]
3     , [nombre_completo_contacto]
4     , [ciudad]
5     , [pais]
6     , [region]
7     , [codigo_postal]
8     , [direccion]
9     , [telefono]
10    , [cliente_vp]
11 FROM [DBO_DATAMART_Jardineria].[dbo].[dim_cliente]
12

```

ID_cliente	nombre_cliente	nombre_completo_contacto	ciudad	pais	region	codigo_postal	direccion	telefono	cliente_vp
1	Goldfish Garden	Goldfish Garden Goldfish	San Francisco	USA	NULL	24006	False Street 52 2 A	5556901745	NULL
2	Gardening Associates	Gardening Associates Wright	Miami	USA	Miami	24010	Walle Avenue	5557410345	NULL
3	Garden Valley	Garden Valley Rouse	New York	USA	NULL	85495	Olney Avenue #22	5552323129	NULL
4	Tendo Garden	Tendo Garden Tendo	Miami	USA	NULL	69690	Nail Street #90	5559123210	NULL
5	Lasas S.A.	Lasas S.A. Lasas	Fuenlabrada	Spain	Madrid	28945	C/Leganes 15	34916540145	NULL
6	Beragua	Beragua Bermejo	Madrid	Spain	Madrid	28942	C/Pontor segundo Gatafe	654987321	NULL
7	Club Golf Puerta del Hemo	Club Golf Puerta del Hemo Lopez	Madrid	Spain	Madrid	28930	C/ramoso delgado Madrid	62456810	NULL
8	Naturagua	Naturagua Penglto	Madrid	Spain	Madrid	28547	C/napalrhonda Boadilla	689234750	NULL
9	DanDistribuciones	DanDistribuciones Serrano	Madrid	Spain	Madrid	28945	C/raones Fuenlabrada	675530001	NULL
10	Madriela de regos	Madriela de regos Tacaño	Madrid	Spain	Madrid	28943	C/Lagañas Fuenlabrada	655983045	NULL
11	Lasas S.A.	Lasas S.A. Lasas	Fuenlabrada	Spain	Madrid	28945	C/Leganes 15	34916540145	NULL
12	Camunas Jardines S.L.	Camunas Jardines S.L. Camunas	San Lorenzo del Escorial	Spain	Madrid	28145	C/Vigenes 45 C/Princesas 2 1B	34914873241	NULL
13	Dardena S.A.	Dardena S.A. Rodriguez	Madrid	Spain	Madrid	28003	C/Nueva York 74	34917453217	NULL
14	Jardin de Flores	Jardin de Flores Villar	Madrid	Spain	Madrid	28950	C/Olea 34	654855643	NULL
15	Flora Manvi	Flora Manvi Rodriguez	Fuenlabrada	Spain	Madrid	28945	C/Leganes24	666555444	NULL
16	Flowers S.A.	Flowers S.A. Fernandez	Montornes del valles	Spain	Barcelona	24586	C/Lux Salzufla	690754159	NULL
17	Naturagarden	Naturagarden Cruz	Madrid	Spain	Madrid	28011	Plaza Magallón 15	612343529	NULL
18	Golf S.A.	Golf S.A. Martinez	Santa cruz de Tenerife	Spain	Islas Canarias	38297	C/Estancado	916458762	NULL
19	Aerotech Golf Management SL	Aerotech Golf Management SL Suarez	Barcelona	Spain	Cataluña	12320	C/Lutardo	964493072	NULL

```

1 SELECT TOP (1000) [ID_empleado]
2     , [nombre_completo]
3     , [puesto]
4     , [ID_jefe]
5     , [ID_oficina]
6 FROM [DBO_DATAMART_Jardineria].[dbo].[dim_empleado]
7

```

ID_empleado	nombre_completo	puesto	ID_jefe	ID_oficina
1	Marcos Magaña Perez	Director General	NULL	8
2	Ruben López Martinez	Subdirector Marketing	1	8
3	Alberto Soria Carrasco	Subdirector Ventas	2	8
4	Maria Solis Jerez	Secretaria	2	8
5	Felipe Rosas Marquez	Representante Ventas	3	8
6	Juan Carlos Ortiz Serrano	Representante Ventas	3	8
7	Carlos Soria Jimenez	Director Oficina	3	4
8	Mariano López Murcia	Representante Ventas	7	4
9	Lucio Campoamor Martín	Representante Ventas	7	4
10	Hilano Rodriguez Huertas	Representante Ventas	7	4
11	Emmanuel Magaña Perez	Director Oficina	3	1
12	José Manuel Martínez De la Osa	Representante Ventas	11	1
13	David Palma Acetuno	Representante Ventas	11	1
14	Oscar Palma Acetuno	Representante Ventas	11	1
15	Francois Fignon	Director Oficina	3	5
16	Lionel Narvaez	Representante Ventas	15	5
17	Laurent Sema	Representante Ventas	15	5
18	Michael Bolton	Director Oficina	3	6
19	Walter Santiago Sanchez Lopez	Representante Ventas	18	6

Conclusiones

1. El proceso ETL es fundamental para garantizar que los datos del Data Mart estén integrados, limpios y listos para su análisis.
2. SSIS provee herramientas visuales robustas para construir flujos de datos eficientes, aunque es importante gestionar correctamente metadatos y relaciones entre tablas para evitar errores durante la carga.
3. La integridad referencial debe verificarse previamente para evitar conflictos al momento de insertar en la tabla de hechos.
4. La identificación de errores como claves foráneas inexistentes o problemas de metadatos permite mejorar progresivamente la calidad del proceso.

Bibliografía

- Microsoft Docs. (2024). *SQL Server Integration Services (SSIS)*. Recuperado de: <https://learn.microsoft.com/es-es/sql/integration-services/>
- Kimball, R., & Ross, M. (2013). *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. Wiley.
- Coronel, C., & Morris, S. (2017). *Database Systems: Design, Implementation, & Management*. Cengage Learning.
- Garcia-Molina, H., Ullman, J., & Widom, J. (2008). *Database Systems: The Complete Book*. Pearson.