

Evidencia de aprendizaje 3.
Proceso de transformación de datos y carga en el data mart final

Autores:

Jose Mauricio Camacho Bravo

PREICA2401B010093

Francisco Luis Cequeda Álvarez

PREICA2401B010094

Tutor:

Victor Hugo Mercado

Ingeniería De Software y Datos

IUDigital de Antioquia

Mayo de 2024

Medellín

Introducción

Durante el proceso de desarrollo de esta actividad perteneciente a la asignatura Base de datos II, se realizará la **creación de dimensiones** y el proceso de **Transformación de datos** desde las tablas de la base de datos **STAGING**, se utilizará la herramienta **Integration Services**, la cual permite **crear las dimensiones, transformar los datos y crear los registros** en la base de datos **STAGING**.

Objetivo

El objetivo de la actividad es crear las dimensiones propuestas en la actividad 1 y a su vez realizar el proceso de transformación de datos a partir de las tablas ya migradas y que existen en la base de datos **STAGING**, este proceso de ETL debe ser realizado por medio de la herramienta **Integration services**.

Planteamiento del problema

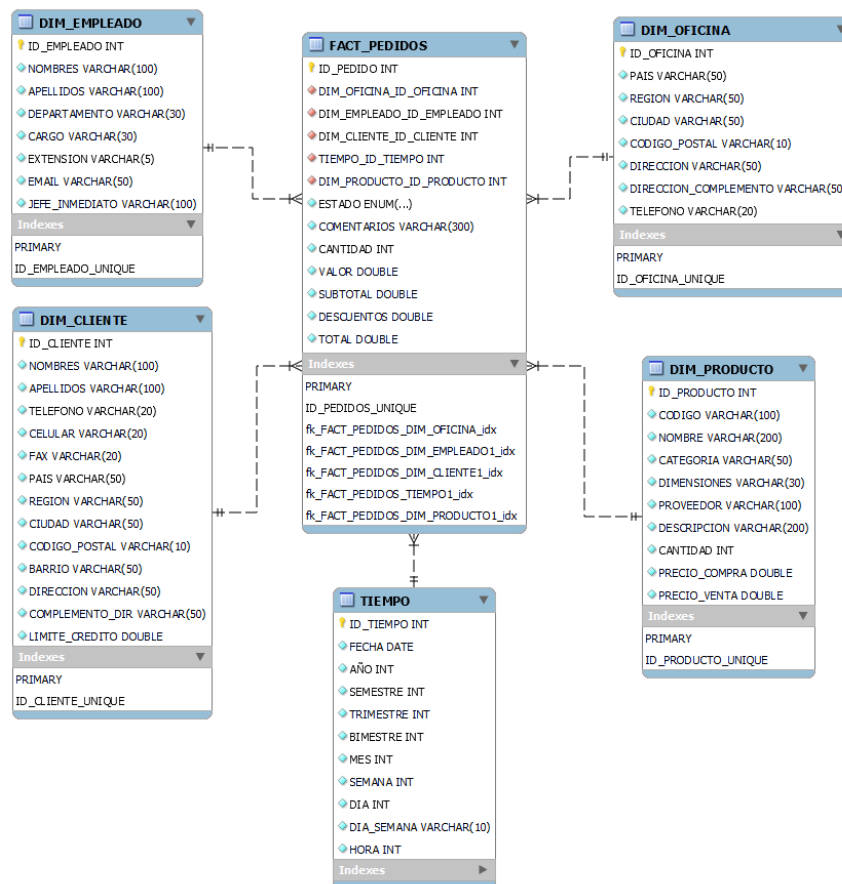
Construir un modelo de extracción de datos, transformación de datos y creación de dimensiones en la base de datos de Staging, este debe permitir transformar los datos de las tablas y posteriormente crear las dimensiones en dicha base de datos (**Staging**), para así realizar el respectivo análisis por medio de herramientas de inteligencia de negocios.

Análisis del problema

Se deben crear las dimensiones de datos a partir de las tablas alojadas en la base de datos Staging, este proceso se debe realizar mediante la herramienta **Integration Services**, la cual es una extensión de visual studio y permite realizar procesos de extracción de datos, transformación de datos y carga de datos, para esto se deben configurar o parametrizar las conexiones a las bases de datos dentro de la herramienta Integration Services.

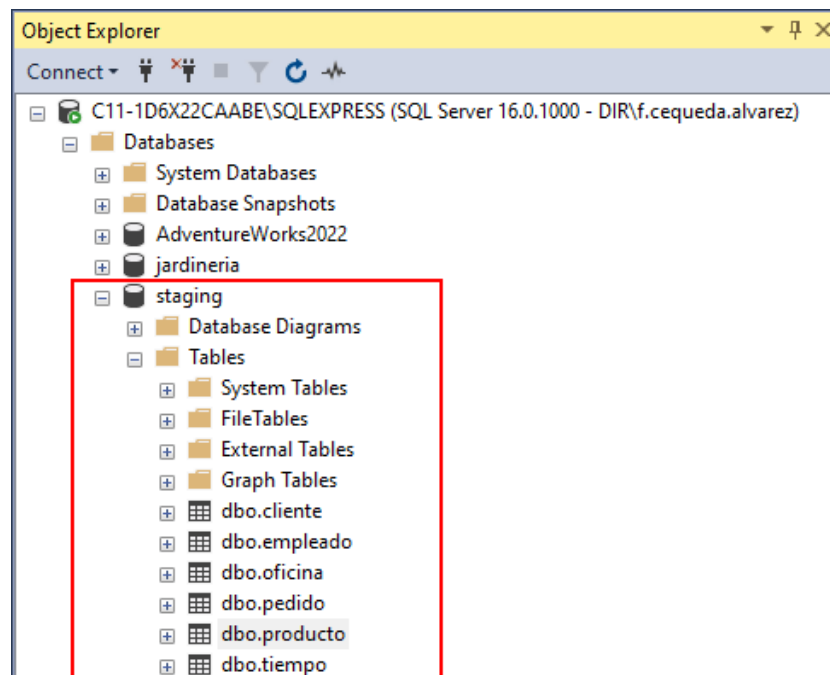
Posterior a la extracción de datos realizada en las actividades anteriores se procede a transformar y crear las dimensiones a partir de las tablas existentes en dicha base de datos.

A continuación se adjunta el modelo inicial propuesto en la actividad 1, el cual sirve como guía para la creación de las dimensiones en la base de datos.



Disponibilidad y consistencia BBDD Staging

Actualmente la base de datos Staging cuenta con las siguientes tablas: **Cliente, Empleado, Oficina, Pedido, Producto y Tiempo**



1 cliente	2 empleado	3 oficina	4 pedido	5 producto	6 tiempo
<ul style="list-style-type: none">id_clienteid_cliente_originnombre_clientenombre_contactoapellido_contactotelefonopaisregionciudadcodigo_postaldirecciondireccion_compllimit_creditoid_emp_rep_ventas	<ul style="list-style-type: none">id_empleadoid_empleado_originnombreapellido1apellido2cargoemailextensionid_oficinaid_jefe	<ul style="list-style-type: none">id_oficinaid_oficina_origindescripcionpaisregionciudadcodigo_postaldirecciondireccion_compltelefono	<ul style="list-style-type: none">id_pedidoid_pedido_origid_productoid_clienteestadocomentariosnumero_lineacantidadprecio_unidadtotalfecha_pedidofecha_esperadafecha_entrega	<ul style="list-style-type: none">id_productoid_producto_origincod_productonombrecategoriacategoria_descdimensionesproveedordescripcioncant_stockprecio_ventaprecio_compra	<ul style="list-style-type: none">idfecha_pedido

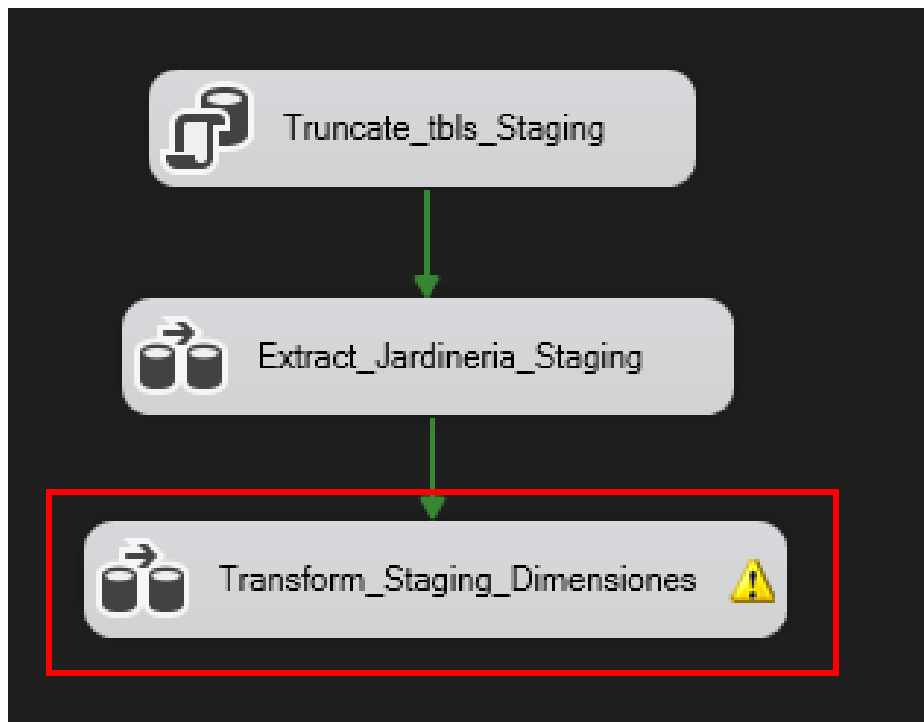
Estas tablas, sus atributos y registros serán primordiales para realizar de manera exitosa el proceso de transformación de datos y creación de las dimensiones para esto se continuará con la **creación de flujos de datos en Integration Services**

Creación de flujos de datos en Integration Services

Los flujos de datos en la herramienta nos permitirán crear tareas automatizadas que utilicen con fuente de datos (origen) la base de datos **staging** y sus respectivas tablas, esto para poder mapear y transformar los datos existentes y que serán insertados en las dimensiones anteriormente descritas.

Flujo Dimensiones:

A continuación se adjunta imagen del flujo de datos principal, el cual contiene como origen a las tablas de la base de datos staging y destino a las dimensiones que serán creadas en la misma base de datos:



Dimensión Tiempo:

Aquí se observa la primera dimensión, esta es creada a partir de las fechas de la tabla pedido:

SQL Query (log) - C1...pedidoahorro (53) X

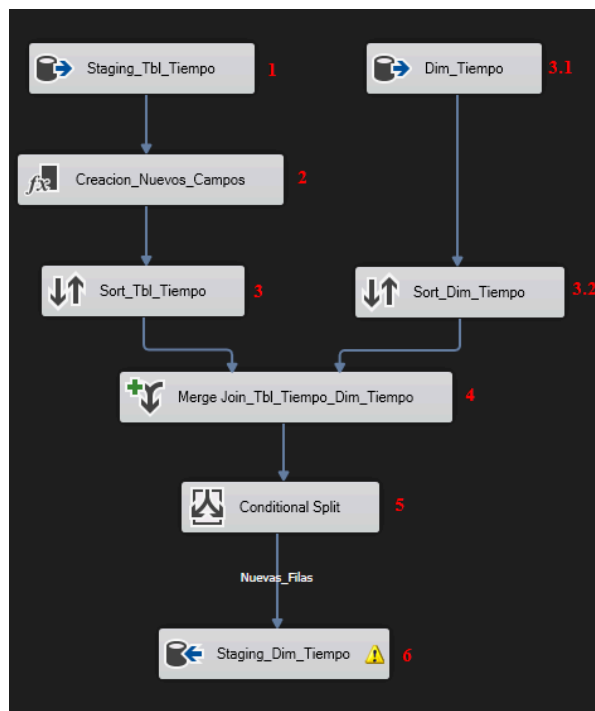
```

SELECT TOP (1000) [id_pedido]
,[id_pedido_orig]
,[id_producto]
,[id_cliente]
,[estado]
,[comentarios]
,[numero_linea]
,[cantidad]
,[precio_unidad]
,[total]
,[fecha_pedido]
,[fecha_esperada]
,[fecha_entrega]
FROM [staging].[dbo].[pedido]

```

100 % Results Messages

	id_pedido	id_pedido_orig	id_producto	id_cliente	estado	comentarios	numero_linea	cantidad	precio_unidad	total	fecha_pedido	fecha_esperada	fecha_entrega
1	1		87	5	Entregado	Pagado a plazos	3	10	70.00	700.00	2006-01-17	2006-01-19	2006-01-19
2	2		151	5	Entregado	Pagado a plazos	1	40	4.00	160.00	2006-01-17	2006-01-19	2006-01-19
3	3		165	5	Entregado	Pagado a plazos	2	25	4.00	100.00	2006-01-17	2006-01-19	2006-01-19
4	4		265	5	Entregado	Pagado a plazos	4	15	19.00	285.00	2006-01-17	2006-01-19	2006-01-19
5	5		276	5	Entregado	Pagado a plazos	5	23	14.00	322.00	2006-01-17	2006-01-19	2006-01-19
6	6		57	5	Entregado	La entrega llego antes de lo esperado	6	3	29.00	87.00	2007-10-23	2007-10-28	2007-10-26
7	7		58	5	Entregado	La entrega llego antes de lo esperado	7	7	8.00	56.00	2007-10-23	2007-10-28	2007-10-26
8	8		164	5	Entregado	La entrega llego antes de lo esperado	3	50	4.00	200.00	2007-10-23	2007-10-28	2007-10-26
9	9		165	5	Entregado	La entrega llego antes de lo esperado	2	20	5.00	100.00	2007-10-23	2007-10-28	2007-10-26
10	10		183	5	Entregado	La entrega llego antes de lo esperado	5	12	6.00	72.00	2007-10-23	2007-10-28	2007-10-26
11	11		251	5	Entregado	La entrega llego antes de lo esperado	1	67	64.00	4288.00	2007-10-23	2007-10-28	2007-10-26
12	12		271	5	Entregado	La entrega llego antes de lo esperado	4	5	462.00	2310.00	2007-10-23	2007-10-28	2007-10-26
13	13		66	5	Rechazado	Limite de credito superado	6	120	9.00	1080.00	2008-06-20	2008-06-25	NULL
14	14		146	5	Rechazado	Limite de credito superado	4	32	5.00	160.00	2008-06-20	2008-06-25	NULL
15	15		147	5	Rechazado	Limite de credito superado	5	11	5.00	55.00	2008-06-20	2008-06-25	NULL
16	16		237	5	Rechazado	Limite de credito superado	1	30	266.00	7980.00	2008-06-20	2008-06-25	NULL
17	17		241	5	Rechazado	Limite de credito superado	2	15	65.00	975.00	2008-06-20	2008-06-25	NULL
18	18		242	5	Rechazado	Limite de credito superado	3	24	25.00	600.00	2008-06-20	2008-06-25	NULL
19	19		48	5	Pendiente	NULL	7	12	8.00	96.00	2009-01-20	2009-01-26	NULL
20	20		51	5	Pendiente	NULL	6	42	8.00	336.00	2009-01-20	2009-01-26	NULL
21	21		58	5	Pendiente	NULL	8	42	9.00	378.00	2009-01-20	2009-01-26	NULL
22	22		176	5	Pendiente	NULL	5	3	6.00	18.00	2009-01-20	2009-01-26	NULL
23	23		179	5	Pendiente	NULL	3	4	6.00	24.00	2009-01-20	2009-01-26	NULL
24	24		180	5	Pendiente	NULL	4	17	9.00	153.00	2009-01-20	2009-01-26	NULL
25	25		181	5	Pendiente	NULL	2	38	10.00	380.00	2009-01-20	2009-01-26	NULL
26	26		246	5	Pendiente	NULL	1	21	59.00	1239.00	2009-01-20	2009-01-26	NULL
27	27		23	1	Pendiente	El cliente requiere que el pedido se l...	1	3	11.00	33.00	2009-01-20	2009-01-27	NULL



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla pedido y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de las datos existentes (fechas) se crean nuevos campos para ser insertados en la dimensión tiempo, los cuales pueden ser: Año, Mes, Día, Semana, Trimestre, Día de la Semana.

Derived Column Name	Derived Column	Expression
año	<add as new column>	YEAR(fecha_pedido)
mes	<add as new column>	MONTH(fecha_pedido)
día	<add as new column>	DAY(fecha_pedido)
semana	<add as new column>	DATEPART("wk",fecha_pedido)
trimestre	<add as new column>	DATEPART("qq",fecha_pedido)
día_semana	<add as new column>	DATEPART("dw",fecha_pedido)

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión tiempo
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión tiempo y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), Ordenamiento y Condición.

A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "dim_tiempo" ("id" int,"fecha_pedido" date, "año" int, "mes" int, "dia" int, "semana" int, "trimestre" int, "dia_semana" int, "id_dim" int)
```

Dimensión Producto:

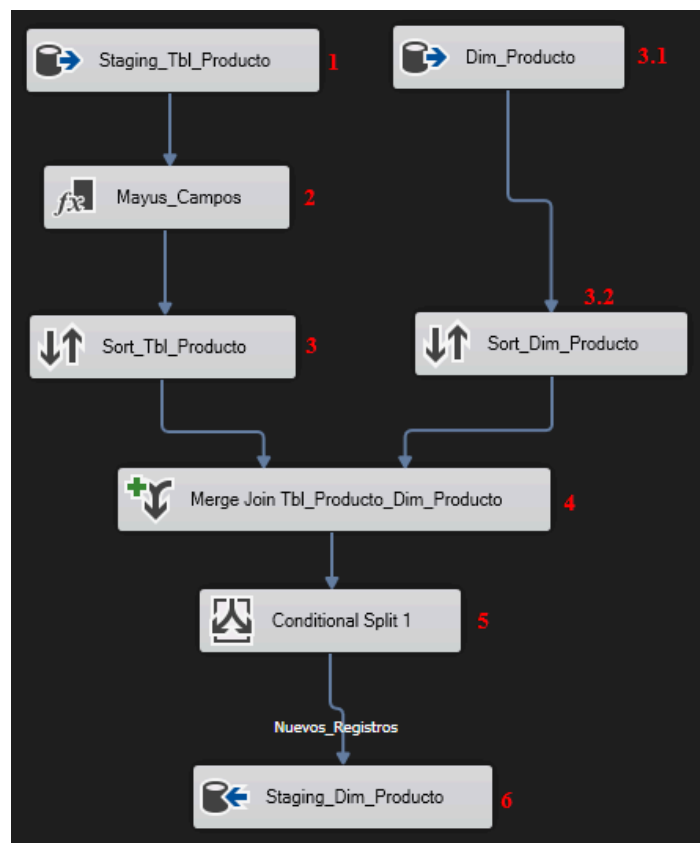
Esta es creada a partir de la tabla producto:

SQLQuery3.sql - C:\...queda.alvarez (56) X

```
SELECT TOP (1000) [id_producto]
,[id_producto_origen]
,[cod_producto]
,[nombre]
,[categoria]
,[categoria_desc]
,[dimensiones]
,[proveedor]
,[descripcion]
,[cant_stock]
,[precio_venta]
,[precio_compra]
FROM [staging].[dbo].[producto]
```

100 % Results Messages

	id_producto	id_producto_origen	cod_producto	nombre	categoria	categoria_desc	dimensiones	proveedor	descripcion	cant_stock	precio_venta
1	1	1	11679	Sierra de Poda 400MM	Herramientas	2 para todo tipo de acción	0,258	HiperGarden Tools	Gracias a la poda se consigue manipular un poco la...	15	14.00
2	2	2	21636	Pala	Herramientas	2 para todo tipo de acción	0,156	HiperGarden Tools	Palas de acero con cresta de corte en la punta para...	15	14.00
3	3	3	22225	Rastrillo de Jardín	Herramientas	2 para todo tipo de acción	1,064	HiperGarden Tools	Fabuloso rastillo que le ayudará a eliminar piedras, h...	15	12.00
4	4	4	30310	Azadón	Herramientas	2 para todo tipo de acción	0,168	HiperGarden Tools	Longitud 24cm. Herramienta fabricada en acero y pi...	15	12.00
5	5	5	AR-001	Ajedrea	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Planta aromática que fresca se utiliza para condime...	140	1.00
6	6	6	AR-002	Lavándula Dentata	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Espiglo de jardín. Alhucema rizada, Alhucema dent...	140	1.00
7	7	7	AR-003	Mejorana	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Origanum majorana. No hay que confundirlo con el ...	140	1.00
8	8	8	AR-004	Melissa	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Es una planta perenne (dura varios años) conocida ...	140	1.00
9	9	9	AR-005	Mertha Sativa	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	¿Quién no conoce la Hierbabuena? Se trata de una...	140	1.00
10	10	10	AR-006	Petroselinum Hortense (Peregr)	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Nombre científico o latino: Petroselinum hortense, P...	140	1.00
11	11	11	AR-007	Salvia Mlx	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	La Salvia es un pequeño arbusto que llega hasta el ...	140	1.00
12	12	12	AR-008	Thymus Citrodra (Tomillo Imón)	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Nombre común o vulgar: Tomillo, Trenoncillo Famili...	140	1.00
13	13	13	AR-009	Thymus Vúlgaris	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons	Nombre común o vulgar: Tomillo, Trenoncillo Famili...	140	1.00
14	14	14	AR-010	Santolina Chamaecyparys	Aromaticas	Plantas aromáticas	15-20	Murcia Seasons		140	1.00
15	15	15	FR-1	Expositor Citricos Mlx	Frutales	Árboles pequeños de producción frutal	100-120	Frutales Talavera S.A		15	7.00
16	16	16	FR-10	Limonero 2 años injerto	Frutales	Árboles pequeños de producción frutal		NaranjasValencianas.com	El limonero, pertenece al grupo de los cítricos, tenie...	15	7.00
17	17	17	FR-100	Nectarina	Frutales	Árboles pequeños de producción frutal	8/10	Frutales Talavera S.A	Se trata de un árbol derivado por mutación de los m...	50	11.00
18	18	18	FR-101	Nogal	Frutales	Árboles pequeños de producción frutal	8/10	Frutales Talavera S.A		50	13.00



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla producto y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de los datos existentes se normalizan los campos al momento de insertados en la dimensión producto, a estos se les aplica la función UPPER que convierte el texto en mayúscula.

Derived Column Name	Derived Column	Expression	Data Type	Length
nombre	Replace 'nombre'	UPPER(nombre)	Unicode string [DT_WS...	70
categoria	Replace 'categoria'	UPPER(categoria)	Unicode string [DT_WS...	50
categoria_desc	Replace 'categoria_desc'	UPPER((DT_WSTR,9999)categoria_desc)	Unicode text stream [D...	
proveedor	Replace 'proveedor'	UPPER(proveedor)	Unicode string [DT_WS...	50
descripcion	Replace 'descripcion'	UPPER((DT_WSTR,9999)descripcion)	Unicode text stream [D...	

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión producto.
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión producto y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), **Ordenamiento** y **Condición**.

A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "dim_producto" ("id_producto" int, "cod_producto" nvarchar(15), "nombre" nvarchar(70), "categoria" nvarchar(50), "categoria_desc" nvarchar(max), "dimensiones" nvarchar(25), "proveedor" nvarchar(50), "descripcion" nvarchar(max), "cant_stock" smallint, "precio_venta" numeric(15,2), "precio_compra" numeric(15,2), "id_producto_dim" int)
```

Dimensión Cliente:

Esta es creada a partir de la tabla Cliente:

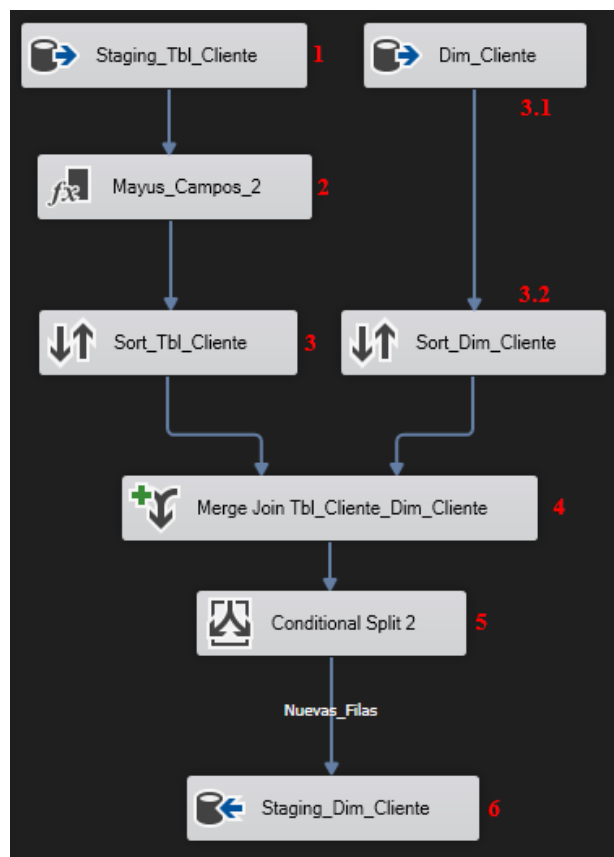
SQLQuery6.sql - C1...queda.alvarez (52) SQLQuery5.sql - C1...queda.alvarez (56)

```
SELECT TOP (1000) [id_cliente]
      ,[id_cliente_origin]
      ,[nombre_cliente]
      ,[nombre_contacto]
      ,[apellido_contacto]
      ,[telefono]
      ,[pais]
      ,[region]
      ,[ciudad]
      ,[codigo_postal]
      ,[direccion]
      ,[direccion_compl]
      ,[limit_credito]
      ,[id_emp_rep_ventas]
FROM [staging].[dbo].[cliente]
```

100 %

Results Messages

	id_cliente	id_cliente_origin	nombre_cliente	nombre_contacto	apellido_contacto	telefono	pais	region	ciudad	codigo_postal	direccion	direccion_compl	limit_credito	id_emp_rep_ventas
1	1	1	GoldFish Garden	Daniel G	GoldFish	5556901745	USA	NULL	San Francisco	24006	False Street 52 2 A	NULL	3000.00	19
2	2	2	Gardening Associates	Anne	Wright	5557410345	USA	Miami	Miami	24010	Wall-e Avenue	NULL	6000.00	19
3	3	3	Genudo Valley	Link	Flaute	5552323129	USA	NULL	New York	85495	Oaks Avenue n°22	NULL	12000.00	22
4	4	4	Tendo Garden	Akane	Tendo	55591233210	USA	NULL	Miami	696969	Null Street n°69	NULL	600000.00	22
5	5	5	Lasas S.A.	Antonio	Lasas	34916540145	Spain	Madrid	Fuenlabrada	28945	C/Leganes 15	NULL	154310.00	8
6	6	6	Beragua	Jose	Bemejo	654987321	Spain	Madrid	Madrid	28942	C/pintor segundo	Getafe	20000.00	11
7	7	7	Club Golf Puerta del hierro	Paco	Lopez	62456810	Spain	Madrid	Madrid	28930	C/sinesio delgado	Madrid	40000.00	11
8	8	8	Naturagua	Guillermo	Rengifo	689234750	Spain	Madrid	Madrid	28947	C/mojadahonda	Boadilla	32000.00	11
9	9	9	DaraDistribuciones	Jose	Serrano	675598001	Spain	Madrid	Madrid	28946	C/azores	Fuenlabrada	50000.00	11
10	10	10	Madriña de riegos	Jose	Tacaño	655983045	Spain	Madrid	Madrid	28943	C/Lagañas	Fuenlabrada	20000.00	11
11	11	11	Lasas S.A.	Antonio	Lasas	34916540145	Spain	Madrid	Fuenlabrada	28945	C/Leganes 15	NULL	154310.00	8
12	12	12	Camunas Jardines S.L.	Pedro	Camunas	34914873241	Spain	Madrid	San Lorenzo del Escorial	28145	C/Virgenes 45	C/Princesas 2 1ºB	16481.00	8
13	13	13	Dardena S.A.	Juan	Rodriguez	34912453217	Spain	Madrid	Madrid	28003	C/Nueva York 74	NULL	321000.00	8
14	14	14	Jardin de Flores	Javier	Villar	654865643	Spain	Madrid	Madrid	28950	C/ Oña 34	NULL	40000.00	30
15	15	15	Flores Marvi	Maria	Rodriguez	666555444	Spain	Madrid	Fuenlabrada	28945	C/Leganes24	NULL	1500.00	5
16	16	16	Flowers, S.A	Beatriz	Fernandez	698754159	Spain	Barcelona	Montornes del valles	24586	C/Luis Salquillo4	NULL	3500.00	5
17	17	17	Naturajardin	Victoria	Cruz	612343529	Spain	Madrid	Madrid	28011	Plaza Magallón 15	NULL	5050.00	30
18	18	18	Golf S.A.	Luis	Martinez	916458762	Spain	Islas Canarias	Santa cruz de Tenerife	38297	C/Estancado	NULL	30000.00	12
19	19	19	Americh Golf Management SL	Mario	Suarez	964493072	Spain	Cataluña	Barcelona	12320	C/Letardo	NULL	20000.00	12
20	20	20	Aloha	Cristian	Rodriguez	916485852	Spain	Canarias	Canarias	35488	C/Roman 3	NULL	50000.00	12
21	21	21	El Prat	Francisco	Camacho	916882323	Spain	Cataluña	Barcelona	12320	Avenida Tibidabo	NULL	30000.00	12
22	22	22	Sotogrande	Maria	Santillana	915576622	Spain	Cadiz	Sotogrande	11310	C/Paseo del Par...	NULL	60000.00	12



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla cliente y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de los datos existentes se normalizan los campos al insertarlos en la dimensión **Cliente**, se les aplica la función **UPPER** que convierte el texto en mayúsculas y se reemplazan nulos con la función **REPLACENULL**.

Derived Column Name	Derived Column	Expression	Data Type	Length
nombre_cliente	Replace 'nombre_clien...	UPPER(nombre_cliente)	Unicode string [DT_WS...	50
nombre_contacto	Replace 'nombre_cont...	UPPER(nombre_contacto)	Unicode string [DT_WS...	30
apellido_contacto	Replace 'apellido_cont...	UPPER(apellido_contacto)	Unicode string [DT_WS...	30
pais	Replace 'pais'	UPPER(pais)	Unicode string [DT_WS...	50
region	Replace 'region'	UPPER(REPLACENULL(region,"-"))	Unicode string [DT_WS...	50
ciudad	Replace 'ciudad'	UPPER(ciudad)	Unicode string [DT_WS...	50
direccion	Replace 'direccion'	UPPER(direccion)	Unicode string [DT_WS...	50
direccion_compl	Replace 'direccion_co...	UPPER(REPLACENULL(direccion_compl,"-"))	Unicode string [DT_WS...	50

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión cliente.
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión cliente y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), **Ordenamiento** y **Condición**.

A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "dim_cliente" ("id_cliente" int, "nombre_cliente" nvarchar(50),
"nombre_contacto" nvarchar(30), "apellido_contacto" nvarchar(30), "telefono"
varchar(15), "pais" nvarchar(50), "region" nvarchar(50), "ciudad" nvarchar(50),
"codigo_postal" nvarchar(10), "direccion" nvarchar(50), "direccion_compl" nvarchar(50),
"limit_credito" numeric(15,2), "id_emp_rep_ventas" int, "id_cliente_dim" int)
```

Dimensión Empleado:

Esta es creada a partir de la tabla Empleado:

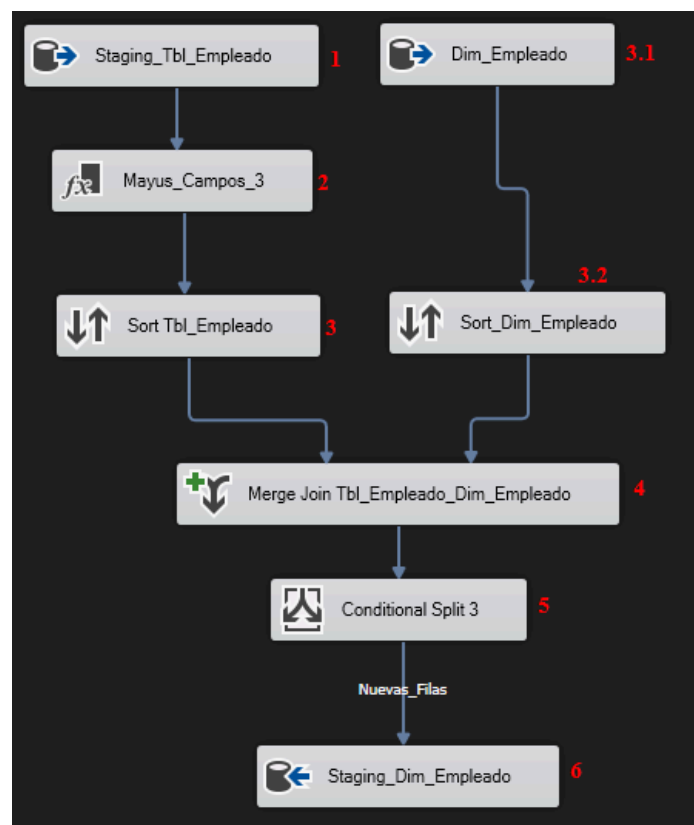
SQLQuery8.sql - C1...queda.alvarez (54)

```
SELECT TOP (1000) [id_empleado]
, [id_empleado_origin]
, [nombre]
, [apellido1]
, [apellido2]
, [cargo]
, [email]
, [extension]
, [id_oficina]
, [id_jefe]
FROM [staging].[dbo].[empleado]
```

100 %

Results Messages

	id_empleado	id_empleado_origin	nombre	apellido1	apellido2	cargo	email	extension	id_oficina	id_jefe
1	1	1	Marcos	Magaña	Perez	Director General	marcos@jardineria.es	3897	8	NULL
2	2	2	Ruben	López	Martinez	Subdirector Marketing	rlopez@jardineria.es	2899	8	1
3	3	3	Alberto	Soria	Carrasco	Subdirector Ventas	asoria@jardineria.es	2837	8	2
4	4	4	Maria	Solís	Jerez	Secretaria	msolis@jardineria.es	2847	8	2
5	5	5	Felipe	Rosas	Marquez	Representante Ventas	frosas@jardineria.es	2844	8	3
6	6	6	Juan Carlos	Ortiz	Serrano	Representante Ventas	cortiz@jardineria.es	2845	8	3
7	7	7	Carlos	Soria	Jimenez	Director Oficina	csoria@jardineria.es	2444	4	3
8	8	8	Mariano	López	Murcia	Representante Ventas	mlopez@jardineria.es	2442	4	7
9	9	9	Lucio	Campoamor	Martín	Representante Ventas	lcampoamor@jardineria.es	2442	4	7
10	10	10	Hilario	Rodriguez	Huertas	Representante Ventas	hrodriguez@jardineria.es	2444	4	7
11	11	11	Emmanuel	Magaña	Perez	Director Oficina	manu@jardineria.es	2518	1	3
12	12	12	José Manuel	Martinez	De la Osa	Representante Ventas	jmmart@hotmail.es	2519	1	11



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla empleado y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de los datos existentes se normalizan los campos al insertarlos en la dimensión **Empleado**, se les aplica la función **UPPER** que convierte el texto en mayús.

Derived Column Name	Derived Column	Expression	Data Type	Length
nombre	Replace 'nombre'	UPPER(nombre)	Unicode string [DT_WSTR]	50
apellido1	Replace 'apellido1'	UPPER(apellido1)	Unicode string [DT_WSTR]	50
apellido2	Replace 'apellido2'	UPPER(apellido2)	Unicode string [DT_WSTR]	50
cargo	Replace 'cargo'	UPPER(cargo)	Unicode string [DT_WSTR]	50
email	Replace 'email'	UPPER(email)	Unicode string [DT_WSTR]	100

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión empleado
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión empleado y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), **Ordenamiento** y **Condición**.

A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "dim_empleado" ("id_empleado" int, "nombre" nvarchar(50), "apellido1" varchar(50), "apellido2" nvarchar(50), "cargo" nvarchar(50), "email" nvarchar(100), "extension" nvarchar(10), "id_oficina" int, "id_jefe" int, "id_empleado_dim" int)
```

Dimensión Oficina:

Esta es creada a partir de la tabla Oficina:

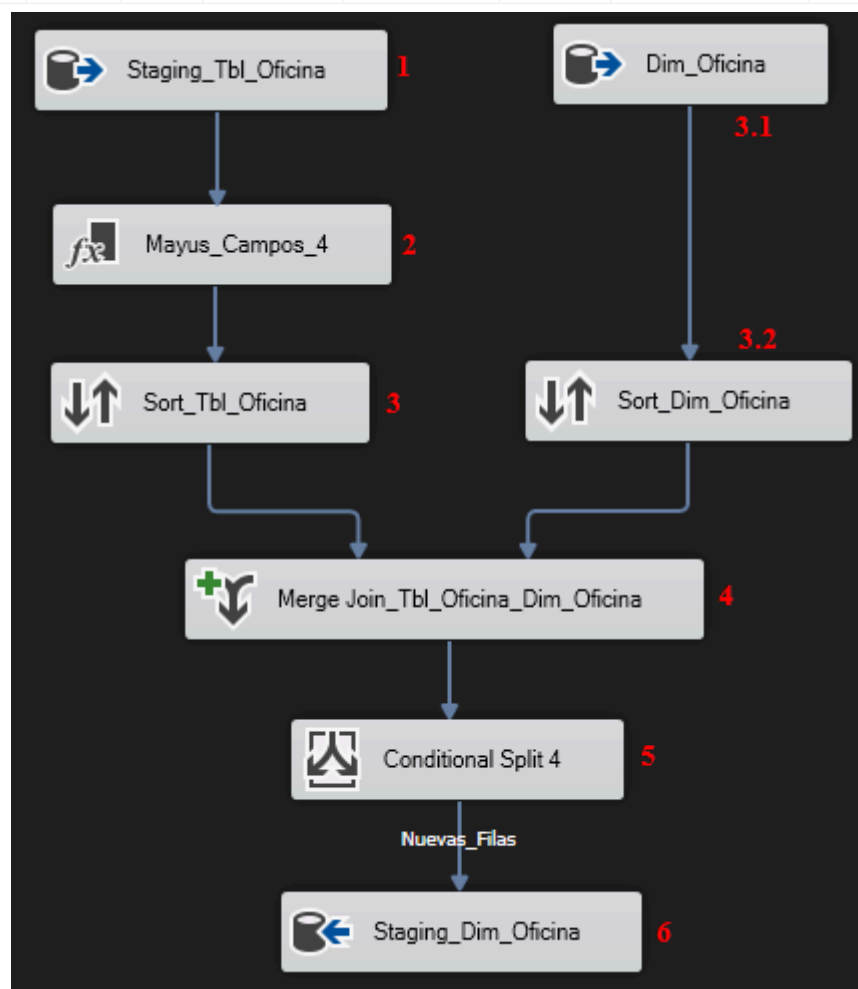
SQLQuery9.sql - C1...queda.alvarez (54)

```
SELECT TOP (1000) [id_oficina]
,[id_oficina_origin]
,[descripcion]
,[pais]
,[region]
,[ciudad]
,[codigo_postal]
,[direccion]
,[direccion_compl]
,[telefono]
FROM [staging].[dbo].[oficina]
```

100 %

Results Messages

	id_oficina	id_oficina_origin	descripcion	pais	region	ciudad	codigo_postal	direccion	direccion_compl	telefono
1	1	1	BCN-ES	España	Barcelona	Barcelona	08019	Avenida Diagonal, 38	3A escalera Derecha	+34 93 3561182
2	2	2	BOS-USA	EEUU	MA	Boston	02108	1550 Court Place	Suite 102	+1 215 837 0825
3	3	3	LON-UK	Inglaterra	EMEA	Londres	EC2N 1HN	52 Old Broad Street	Ground Floor	+44 20 78772041
4	4	4	MAD-ES	España	Madrid	Madrid	28032	Bulevar Indalecio Prieto, 32		+34 91 7514487
5	5	5	PAR-FR	Francia	EMEA	Paris	75017	29 Rue Joffroy d'abbans		+33 14 723 4404
6	6	6	SFC-USA	EEUU	CA	San Francisco	94080	100 Market Street	Suite 300	+1 650 219 4782
7	7	7	SYD-AU	Australia	APAC	Sydney	NSW 2010	5-11 Wentworth Avenue	Floor #2	+61 2 9264 2451
8	8	8	TAL-ES	España	Castilla-LaMancha	Talavera de la Reina	45632	Francisco Aguirre, 32	5º piso (exterior)	+34 925 867231
9	9	9	TOK-JP	Japón	Chiyoda-Ku	Tokyo	102-8578	4-1 Koicho		+81 33 224 5000



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla oficina y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de los datos existentes se normalizan los campos al insertarlos en la dimensión **Oficina**, se les aplica la función **UPPER** que convierte el texto en mayúscula.

Derived Column Name	Derived Column	Expression	Data Type	Length
pais	Replace 'pais'	UPPER(pais)	Unicode string [DT_WSTR]	50
region	Replace 'region'	UPPER(region)	Unicode string [DT_WSTR]	50
ciudad	Replace 'ciudad'	UPPER(ciudad)	Unicode string [DT_WSTR]	30
direccion	Replace 'direccion'	UPPER(direccion)	Unicode string [DT_WSTR]	50
direccion_compl	Replace 'direccion_co...	UPPER(direccion_compl)	Unicode string [DT_WSTR]	50

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión oficina
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión oficina y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), **Ordenamiento** y **Condición**.

A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "dim_oficina" ("id_oficina" int, "descripcion" nvarchar(10), "pais" nvarchar(50), "region" nvarchar(50), "ciudad" nvarchar(30), "codigo_postal" varchar(10), "direccion" nvarchar(50), "direccion_compl" nvarchar(50), "telefono" varchar(20), "id_oficina_dim" int)
```

Dimensión de Hechos (Fact_Pedido):

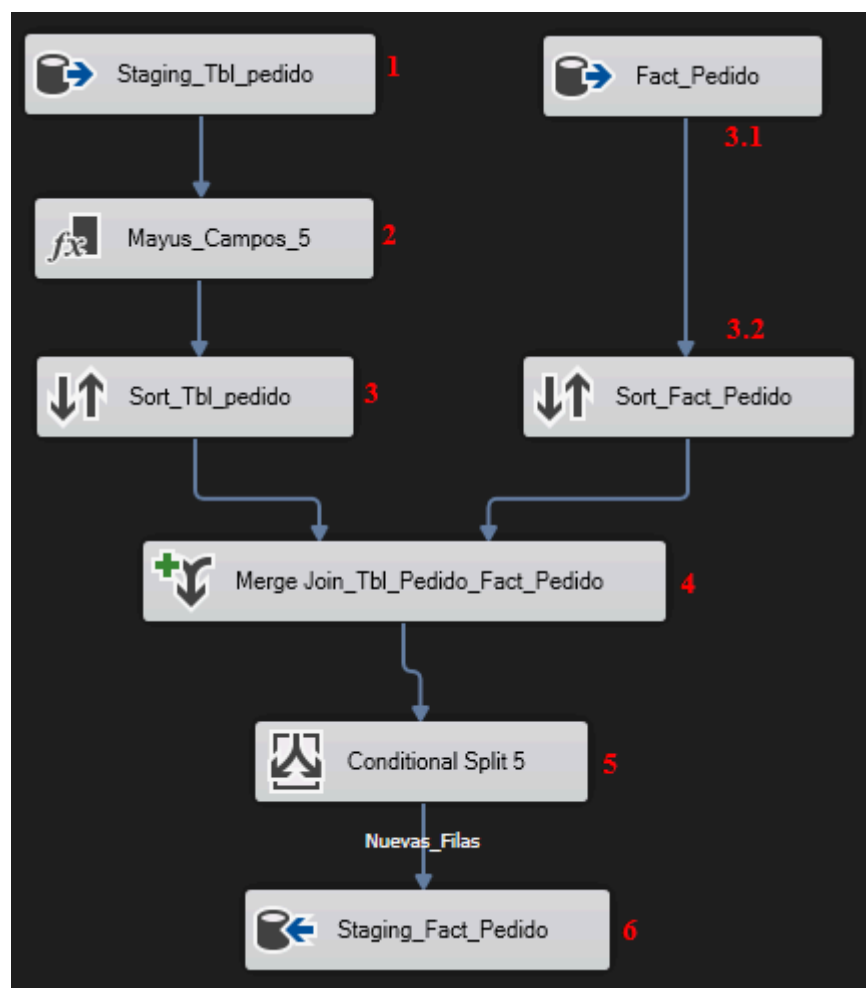
Esta es creada a partir de la tabla pedido:

SQLQuery10.sql - C:\...queda.alvarez (53) SQLQuery9.sql - C:\...queda.alvarez (54)

```
SELECT TOP (1000) [id_pedido]
,[id_pedido_orig]
,[id_producto]
,[id_cliente]
,[estado]
,[comentarios]
,[numero_linea]
,[cantidad]
,[precio_unidad]
,[total]
,[fecha_pedido]
,[fecha_esperada]
,[fecha_entrega]
FROM [staging].[dbo].[pedido]
```

100 %

	id_pedido	id_pedido_orig	id_producto	id_cliente	estado	comentarios	numero_linea	cantidad	precio_unidad	total	fecha_pedido	fecha_esperada	fecha_entrega
1	1	1	87	5	Entregado	Pagado a plazos	3	10	70.00	700.00	2006-01-17	2006-01-19	2006-01-19
2	2	1	151	5	Entregado	Pagado a plazos	1	40	4.00	160.00	2006-01-17	2006-01-19	2006-01-19
3	3	1	165	5	Entregado	Pagado a plazos	2	25	4.00	100.00	2006-01-17	2006-01-19	2006-01-19
4	4	1	265	5	Entregado	Pagado a plazos	4	15	19.00	285.00	2006-01-17	2006-01-19	2006-01-19
5	5	1	276	5	Entregado	Pagado a plazos	5	23	14.00	322.00	2006-01-17	2006-01-19	2006-01-19
6	6	2	57	5	Entregado	La entrega llego antes de lo esperado	6	3	29.00	87.00	2007-10-23	2007-10-28	2007-10-26
7	7	2	58	5	Entregado	La entrega llego antes de lo esperado	7	7	8.00	56.00	2007-10-23	2007-10-28	2007-10-26
8	8	2	164	5	Entregado	La entrega llego antes de lo esperado	3	50	4.00	200.00	2007-10-23	2007-10-28	2007-10-26
9	9	2	165	5	Entregado	La entrega llego antes de lo esperado	2	20	5.00	100.00	2007-10-23	2007-10-28	2007-10-26
10	10	2	183	5	Entregado	La entrega llego antes de lo esperado	5	12	6.00	72.00	2007-10-23	2007-10-28	2007-10-26
11	11	2	251	5	Entregado	La entrega llego antes de lo esperado	1	67	64.00	4288.00	2007-10-23	2007-10-28	2007-10-26
12	12	2	271	5	Entregado	La entrega llego antes de lo esperado	4	5	462.00	2310.00	2007-10-23	2007-10-28	2007-10-26
13	13	3	66	5	Rechazado	Limite de credito superado	6	120	9.00	1080.00	2008-06-20	2008-06-25	NULL
14	14	3	146	5	Rechazado	Limite de credito superado	4	32	5.00	160.00	2008-06-20	2008-06-25	NULL
15	15	3	147	5	Rechazado	Limite de credito superado	5	11	5.00	55.00	2008-06-20	2008-06-25	NULL
16	16	3	237	5	Rechazado	Limite de credito superado	1	30	266.00	7980.00	2008-06-20	2008-06-25	NULL
17	17	3	241	5	Rechazado	Limite de credito superado	2	15	65.00	975.00	2008-06-20	2008-06-25	NULL
18	18	3	242	5	Rechazado	Limite de credito superado	3	24	25.00	600.00	2008-06-20	2008-06-25	NULL
19	19	4	48	5	Pendiente	NULL	7	12	8.00	96.00	2009-01-20	2009-01-26	NULL



En el flujo se observan varios componentes los cuales permiten realizar operaciones diferentes las cuales pueden ser aplicadas a la transformación de datos, comparación de valores u ordenamiento, a continuación se detallan los procesos enumerados:

1. **Definición del origen de los datos:** Aquí se indica como origen tabla pedido y se toman todos los datos.
2. **Creación de nuevos campos:** A partir de los datos existentes se normalizan los campos al insertarlos en la dimensión de **Hechos (Fact_Pedido)**, se les aplica la función **UPPER** que convierte el texto en mayúscula y se reemplazan nulos con la función **REPLACENULL**.

Derived Column Name	Derived Column	Expression	Data Type	Length
estado	Replace 'estado'	UPPER(estado)	Unicode string [DT_WSTR]	15
comentarios	Replace 'comentarios'	UPPER(REPLACENULL((DT_WSTR,9999)comentarios,"-"))	Unicode text stream [DT_NTEXT]	
fecha_entrega	Replace 'fecha_entrega'	REPLACENULL(fecha_entrega,"-")	four-byte signed integer [DT_I4]	

3. **Ordenamiento:** Se ordenan los registros ascendentemente basado en el campo id.
 - 3.1. **Origen dimensión tiempo:** Para poder realizar la comparación de los registros y evitar tener duplicidad de datos, debemos de crear la dimensión de Hechos (Fact_Pedido)
 - 3.2. **Ordenamiento dimensión tiempo:** Posterior a la creación de la dimensión procedemos a ordenar los registros por su identificador.
4. **Merge Join:** Después de haber ordenado los dos orígenes (tabla y dimensión) procedemos a realizar un merge join de tipo **LEFT JOIN**, esto para consultar y verificar que los registros que están en la tabla no estén en la dimensión
5. **Conditional Split:** Este se crea para agregar una condición booleana, la cual valida si los registros de la tabla no están en la dimensión, para su posterior inserción.
6. **Definición Destino:** Aquí se crea la dimensión de Hechos (Fact_Pedido) y se insertan los datos, basado en las tareas anteriores: **Creación de nuevos campos** (Derived Column), Ordenamiento y Condición.

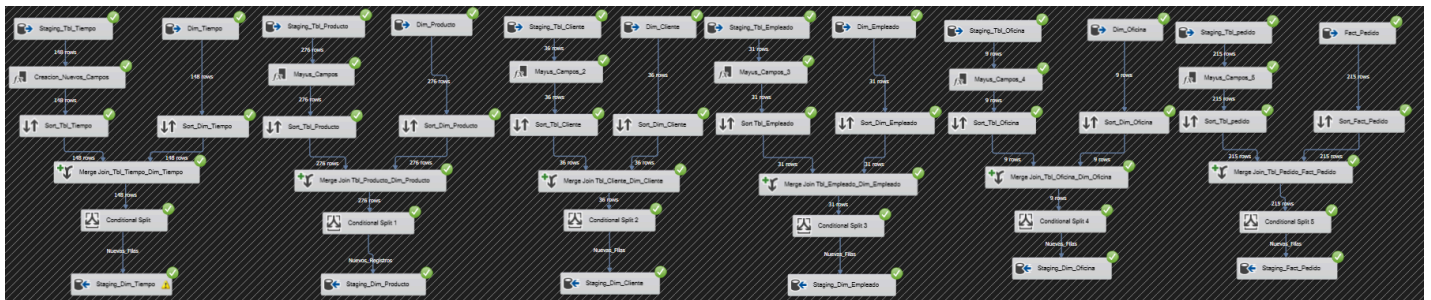
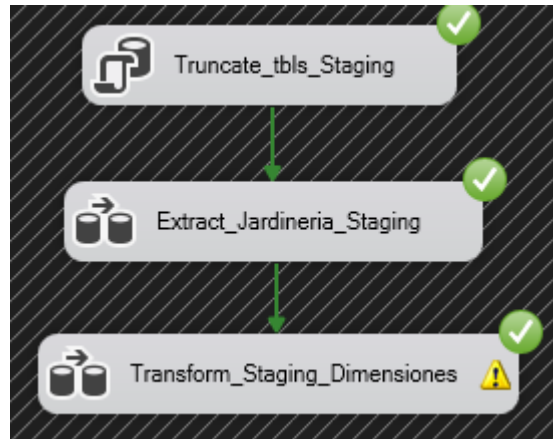
A continuación se adjunta el query que arroja la herramienta para la creación de la dimensión:

Unset

```
CREATE TABLE "fact_pedido" ("id_pedido" int, "id_producto" int, "id_cliente" int, "estado" nvarchar(15), "comentarios" nvarchar(max), "numero_linea" smallint, "cantidad" int, "precio_unidad" numeric(15,2), "total" numeric(26,2), "fecha_pedido" int, "fecha_esperada" int, "fecha_entrega" int, "id_pedido_dim" int)
```

Verificación y ejecución del flujo de datos:

Por último realizamos la ejecución del flujo del modelo de datos, en el cual se definieron las tareas anteriormente mencionadas:



Anexos:

https://drive.google.com/drive/u/2/folders/1IUl0c43m5jQrjXd_SDCHs7IARygPdSh