

Proceso ETL

DOCENTE:

Víctor Hugo Mercado Ramos

ESTUDIANTES:

Camilo Londoño Mesa
José Anderson Castañeda Z.
Daisy Yulieth Jiménez Higueta

CURSO:

Bases de datos II

EVIDENCIA DE APRENDIZAJE:

Proceso ETL

FECHA

30 de marzo de 2025
MEDELLÍN

Introducción

El análisis de datos es una herramienta fundamental en la toma de decisiones empresariales, y su correcta implementación permite obtener información valiosa que puede optimizar procesos y mejorar el rendimiento de una organización. En este trabajo, durante la construcción del Data Mart para la empresa de jardinería, se implementó un proceso ETL (Extracción, Transformación y Carga) para garantizar la integración eficiente de los datos desde la base de datos operativa hasta la estructura dimensional.

Inicialmente, se analizaron los datos almacenados en la base de datos de Jardinería y se diseñó un entorno intermedio de almacenamiento, conocido como base de datos Staging. Este proceso es fundamental dentro de los sistemas de integración de datos, ya que permite la depuración, transformación y estructuración de la información antes de su uso en análisis más avanzados.

Se llevó a cabo una revisión detallada de la base de datos Jardinería para identificar cuáles de sus datos son relevantes y deben trasladarse a la base de datos Staging. Posteriormente, se diseñó la estructura de las tablas que conformarán este nuevo entorno, asegurando que respondan a las necesidades de almacenamiento y procesamiento de la información seleccionada.

Para la migración de datos, se construyeron consultas SQL que permitieron extraer los registros necesarios desde Jardinería y almacenarlos en la base de datos Staging. Se ejecutaron y validaron estas consultas, garantizando la integridad y coherencia de los datos transferidos.

Posteriormente, en la fase de transformación, se realizaron procesos de limpieza, normalización y enriquecimiento de los datos, asegurando la correcta asignación de claves foráneas y la coherencia de la información en las dimensiones. Finalmente, en la etapa de carga, los datos procesados fueron insertados en el modelo dimensional, alimentando las diferentes dimensiones (DimEmpleado, DimTiempo, DimProducto, entre otras) y la tabla de hechos.

Este proceso permitió estructurar un modelo optimizado para análisis, facilitando la generación de reportes y consultas de valor para la toma de decisiones estratégicas en la empresa de jardinería.

Objetivo General

Diseñar e implementar un Data Mart para la empresa de jardinería, aplicando un proceso ETL eficiente que permita la extracción, transformación y carga de datos desde la base de datos transaccional hacia una estructura dimensional optimizada para la toma de decisiones.

Objetivos Específicos

- Analizar la base de datos Jardinería para identificar sus características y los datos relevantes.
- Extraer la información relevante desde la base de datos operativa (Jardinería) y almacenarla en una base de datos intermedia (Staging), asegurando la integridad y consistencia de los datos.
- Transformar los datos mediante procesos de limpieza, normalización y asignación de claves foráneas, garantizando la coherencia entre las tablas de hechos y dimensiones.
- Cargar la información transformada en el modelo dimensional, estructurando correctamente las tablas de hechos y dimensiones (DimEmpleado, DimTiempo, DimProducto, DimCliente)
- Implementar mecanismos de validación y control de calidad para verificar la correcta migración y estructuración de los datos en el Data Mart.
- Desarrollar consultas y reportes que permitan la explotación analítica de la información almacenada en el Data Mart.

Planteamiento del Problema

La empresa de jardinería enfrenta el desafío de analizar una base de datos transaccional que contiene información sobre sus productos, categorías de productos, empleados y ventas. Sin embargo, la estructura actual de la base de datos no está optimizada para el análisis directo, lo que dificulta la obtención de información estratégica para la toma de decisiones.

La base de datos Jardinería almacena grandes volúmenes de datos con redundancias, inconsistencias y estructuras no adecuadas para consultas analíticas eficientes. Esto afecta el rendimiento de los procesos de consulta y análisis, dificultando la identificación de patrones y tendencias clave para la optimización del negocio.

Para abordar este problema, se requiere la implementación de un proceso ETL que permita la extracción, transformación y carga de los datos en un Data Mart diseñado específicamente para la toma de decisiones. La primera etapa consiste en la extracción de datos desde la base de datos Jardinería hacia un entorno intermedio (*Staging*), donde se almacenan y organizan los datos sin afectar la operatividad del sistema transaccional. Luego, en la fase de transformación, se aplican procesos de limpieza, depuración, normalización y asignación de claves foráneas para garantizar la calidad e integridad de los datos. Finalmente, los datos procesados son cargados en el Data Mart, donde se estructuran en un modelo dimensional que permite consultas rápidas y análisis eficientes.

La ausencia de una arquitectura adecuada para la gestión de datos impacta negativamente la eficiencia de los procesos analíticos, aumenta los tiempos de consulta y limita la capacidad de generar reportes estratégicos. Por ello, la

construcción de un **Data Mart** basado en un proceso **ETL** representa una solución integral que mejora la estructuración y disponibilidad de la información, facilitando su explotación para la toma de decisiones basada en datos.

Análisis del problema

La base de datos Jardinería almacena información relevante para la gestión de productos, clientes, pedidos y demás operaciones comerciales. Sin embargo, su estructura actual puede presentar desafíos en términos de eficiencia, integridad y facilidad de análisis. Algunos de los problemas identificados incluyen:

Datos no optimizados: La información almacenada en la base de datos Jardinería está diseñada principalmente para el registro transaccional, lo que puede generar dificultades al momento de realizar consultas analíticas o reportes detallados.

Redundancia e inconsistencias: La presencia de datos repetidos o inconsistentes puede afectar la calidad de la información y comprometer la precisión de los análisis posteriores.

Falta de un entorno de preparación de datos: No contar con una base de datos intermedia (Staging) implica que los datos deben ser transformados directamente en la fuente o en cada consulta, lo que puede ser ineficiente y generar errores.

Rendimiento y tiempos de respuesta: Consultas sobre grandes volúmenes de datos pueden volverse lentas y afectar el desempeño general del sistema, dificultando la obtención de información en tiempo real.

Ante estos desafíos, la implementación de un Data Mart, alimentado a través de un proceso ETL, permitirá organizar, limpiar y transformar los datos antes de su análisis. Este enfoque garantizará la calidad de la información, optimizará los tiempos de consulta y facilitará la generación de reportes estratégicos y modelos analíticos más precisos y eficientes.

Propuesta de solución

1. Descripción de la base de datos Staging

La base de datos Staging propuesta se diseñó como un entorno intermedio de almacenamiento que facilita la depuración, transformación y estructuración de los datos provenientes de la base de datos Jardinería. Su principal objetivo es optimizar la calidad e integridad de la información, asegurando que esté lista para su posterior análisis y uso en reportes o procesos de inteligencia de negocio.

La base de datos Staging se compone de un conjunto de tablas que han sido diseñadas para almacenar únicamente los datos relevantes extraídos de

Jardinería, evitando redundancias y mejorando la eficiencia de las consultas. A continuación, se describen las principales tablas:

ClienteST: Contiene información de los clientes, incluyendo su ID_cliente, nombre_cliente, nombre_contacto, telefono, ciudad.

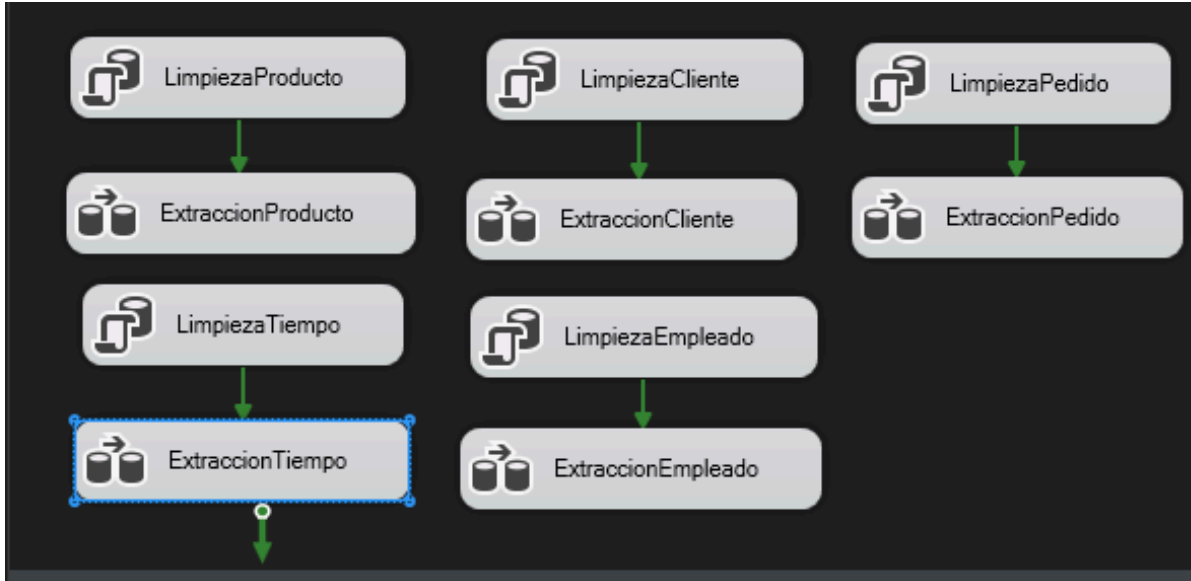
ProductoST: Almacena los productos disponibles en el sistema, con detalles como ID_producto, CodigoProducto, nombre, Categoria, Desc_Categoria, proveedor, dimensiones, descripcion, cantidad_en_stock, precio_venta, precio_proveed.

PedidoST: Registra los pedidos realizados por los clientes, incluyendo datos clave como ID_pedido, ID_cliente, ID_detalle_pedido, ID_producto, cantidad, precio_unidad, numero_linea

EmpleadoST: Contiene información de los empleados, incluyendo su ID_empleado, nombre, apellido1, apellido2, extension, email, puesto, ID_oficina, Descripcion, ciudad, pais.

TiempoST: Contiene información de la fecha de realización de las ventas y su identificación.

2. Diseño de la Base de datos Staging



3. Lista de las consultas utilizadas para traer los registros de Jardinería a la Base de Datos Staging.

Dim_Producto

```
SELECT      ID_producto, CodigoProducto, nombre, Categoria, proveedor,
dimensiones, descripcion, cantidad_en_stock, precio_venta, precio_proveedor
FROM        producto
```

```
CREATE TABLE "ProductoST" (
  "ID_producto" int,
  "CodigoProducto" nvarchar(15),
  "nombre" nvarchar(70),
  "Categoria" int,
  "proveedor" nvarchar(50),
  "dimensiones" nvarchar(25),
  "descripcion" nvarchar(max),
  "cantidad_en_stock" smallint,
  "precio_venta" numeric(15,2),
  "precio_proveedor" numeric(15,2)
)
```

```
SELECT      Id_Categoria, Desc_Categoria
FROM        Categoria_producto
```

```
CREATE TABLE "CategoriaProductoST" (
  "Id_Categoria" int,
  "Desc_Categoria" nvarchar(50),
)
```

Dim_Tiempo

```
SELECT      fecha_pedido, ID_pedido
FROM        pedido
```

```
CREATE TABLE "PedidoST" (
  "fecha_pedido" date,
  "ID_pedido" int
)
```

Dim_Cliente

```
SELECT      ID_cliente, nombre_cliente, nombre_contacto, telefono, ciudad
FROM        cliente
```

```
CREATE TABLE "ClienteST" (
  "ID_cliente" int,
  "nombre_cliente" nvarchar(50),
  "nombre_contacto" nvarchar(30),
```

```
    "telefono" nvarchar(15),  
    "ciudad" nvarchar(50)  
)
```

Dim_Empleado

```
SELECT      ID_empleado, nombre, apellido1, apellido2, extension, email, puesto,  
            ID_oficina  
FROM        empleado
```

```
CREATE TABLE "EmpleadoST" (  
    "ID_empleado" int,  
    "nombre" nvarchar(50),  
    "apellido1" nvarchar(50),  
    "apellido2" nvarchar(50),  
    "extension" nvarchar(10),  
    "email" nvarchar(100),  
    "puesto" nvarchar(50),  
    "ID_oficina" int  
)
```

```
SELECT      ID_oficina, Descripcion, ciudad, pais  
FROM        oficina
```

```
CREATE TABLE "OficinaST" (  
    "ID_oficina" int,  
    "Descripcion" nvarchar(10),  
    "ciudad" nvarchar(30),  
    "pais" nvarchar(50)  
)
```

FAC_Pedido

```
SELECT      ID_pedido, ID_cliente, fecha_pedido
FROM        pedido
```

```
CREATE TABLE "PedidoST" (
  "ID_pedido" int,
  "ID_cliente" int
  "fecha_pedido" date
)
```

```
SELECT      ID_detalle_pedido, ID_pedido, ID_producto, cantidad, precio_unidad,
numero_linea
FROM        detalle_pedido
```

```
CREATE TABLE "DetallePedidoST" (
  "ID_detalle_pedido" int,
  "ID_pedido" int,
  "ID_producto" int,
  "cantidad" int,
  "precio_unidad" numeric(15,2),
  "numero_linea" smallint
)
```

4. Lista de las consultas utilizadas para transformar los registros de la Base de Datos Staging.

DimTiempo


```
SELECT    DISTINCT fecha_pedido
FROM      PedidoST
```

```
CREATE TABLE "DimTiempoST" (
    "fecha_pedido" date,
    "Dia" int,
    "Mes" int,
    "Año" int,
    "Trimestre" int
)
```

DimProducto

```
SELECT          ProductoST.ID_producto,  ProductoST.CodigoProducto,
ProductoST.nombre,      ProductoST.Categoria,      ProductoST.proveedor,
ProductoST.dimensiones, ProductoST.descripcion, ProductoST.cantidad_en_stock,
                  ProductoST.precio_venta, ProductoST.precio_proveedor,
CategoriaProductoST.Desc_Categoria
FROM            CategoriaProductoST CROSS JOIN
                ProductoST
```

```
CREATE TABLE "DimProductoST" (
    "ID_producto" int,
    "CodigoProducto" nvarchar(15),
    "nombre" nvarchar(70),
    "Categoria" int,
    "proveedor" nvarchar(50),
    "dimensiones" nvarchar(25),
    "descripcion" nvarchar(max),
    "cantidad_en_stock" smallint,
    "precio_venta" numeric(15,2),
    "precio_proveedor" numeric(15,2),
    "Desc_Categoria" nvarchar(50)
)
```

DimCliente

```
SELECT    ID_cliente, nombre_cliente, nombre_contacto, telefono, ciudad
FROM      ClienteST
```

```
CREATE TABLE "DimClienteST" (
    "ID_cliente" int,
    "nombre_cliente" nvarchar(50),
    "nombre_contacto" nvarchar(30),
    "telefono" nvarchar(15),
    "ciudad" nvarchar(50)
)
```

DimEmpleado

```
SELECT
    EmpleadoST.ID_empleado, EmpleadoST.nombre,
    EmpleadoST.apellido1, EmpleadoST.apellido2, EmpleadoST.extension,
    EmpleadoST.email, EmpleadoST.puesto, EmpleadoST.ID_oficina,
    OficinaST.ID_oficina AS Expr1,
    OficinaS.Descripción, OficinaST.ciudad, OficinaST.pais
FROM
    EmpleadoST CROSS JOIN
    OficinaST
```

```
CREATE TABLE "DimEmpleadoST" (
    "ID_empleado" int,
    "nombre" nvarchar(50),
    "apellido1" nvarchar(50),
    "apellido2" nvarchar(50),
    "extension" nvarchar(10),
    "email" nvarchar(100),
    "puesto" nvarchar(50),
    "ID_oficina" int,
    "Expr1" int,
    "Descripcion" nvarchar(10),
    "ciudad" nvarchar(30),
    "pais" nvarchar(50),
    "nombrecompleto" nvarchar(50)
)
```

FACPedido

```
SELECT PedidoST.ID_pedido, PedidoST.ID_cliente, DimTiempoST.fecha_pedido,
    DetallePedidoST.ID_pedido, DetallePedidoST.ID_producto,
    DetallePedidoST.cantidad, DetallePedidoST.precio_unidad,
    DetallePedidoST.numero_linea
```

```

FROM [dbo].[PedidoST] INNER JOIN [dbo].[DetallePedidoST] ON
PedidoST.ID_pedido=DetallePedidoST.ID_pedido
INNER JOIN [dbo].[DimTiempoST] ON
PedidoST.fecha_pedido=DimTiempoST.fecha_pedido
INNER JOIN [dbo].[DimProductoST] on
DimProductoST.ID_producto=DetallePedidoST.ID_producto
INNER JOIN [dbo].[DimClienteST] on
DimClienteST.ID_cliente=PedidoST.ID_cliente

```

order by PedidoST.ID_pedido

```

CREATE TABLE "FACPedido" (
  "ID_pedido" int,
  "ID_cliente" int,
  "fecha_pedido" date,
  "ID_producto" int,
  "cantidad" int,
  "precio_unidad" numeric(15,2),
  "numero_linea" smallint
)

```

5. Lista de las consultas utilizadas para cargar los registros transformados al Data Mart.

DimCliente

```
CREATE TABLE "DimCliente" (  
    "ID_cliente" int primary key,  
    "nombre_cliente" nvarchar(50),  
    "nombre_contacto" nvarchar(30),  
    "telefono" nvarchar(15),  
    "ciudad" nvarchar(50)  
)
```

DimTiempo

```
CREATE TABLE "DimTiempo" (  
    "fecha_pedido" date primary key,  
    "Dia" int,  
    "Mes" int,  
    "Año" int,  
    "Trimestre" int  
)
```

DimProducto

```
CREATE TABLE "DimProducto" (  
    "ID_producto" int primary key,  
    "CodigoProducto" nvarchar(15),  
    "nombre" nvarchar(70),  
    "Categoria" int,  
    "proveedor" nvarchar(50),  
    "dimensiones" nvarchar(25),  
    "descripcion" nvarchar(max),  
    "cantidad_en_stock" smallint,  
    "precio_venta" numeric(15,2),  
    "precio_proveedor" numeric(15,2),  
    "Desc_Categoria" nvarchar(50)  
)
```

DimEmpleado

```
CREATE TABLE "DimEmpleado" (  
    "ID_empleado" int primary key,  
    "nombre" nvarchar(50),
```

```
"apellido1" nvarchar(50),  
"apellido2" nvarchar(50),  
"extension" nvarchar(10),  
"email" nvarchar(100),  
"puesto" nvarchar(50),  
"ID_oficina" int,  
"Expr1" int,  
"Descripcion" nvarchar(10),  
"ciudad" nvarchar(30),  
"pais" nvarchar(50),  
"nombrecompleto" nvarchar(50)  
)
```

FACPedido

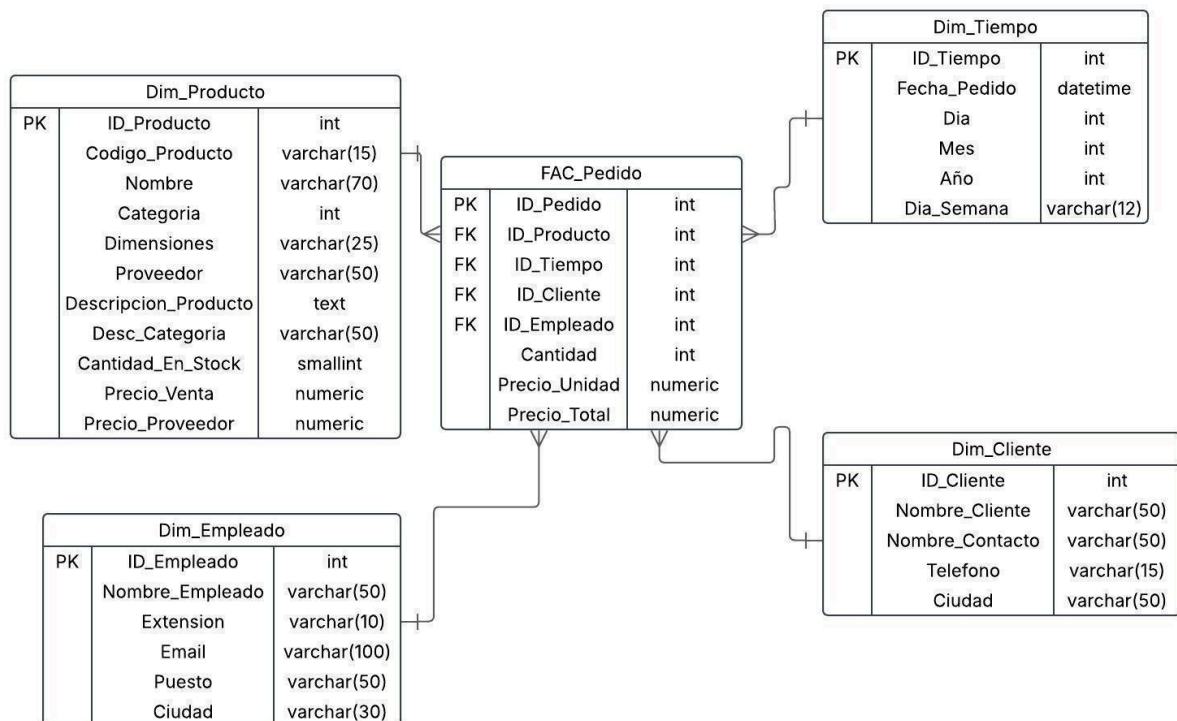
```
CREATE TABLE "FACPedido" (  
    "ID_pedido" int,  
    "ID_cliente" int,  
    "fecha_pedido" date primary key,  
    "ID_producto" int,  
    "cantidad" int,  
    "precio_unidad" numeric(15,2),  
    "numero_linea" smallint  
)
```

5. Conclusiones

- La implementación del Data Mart a partir de la información almacenada en la base de datos Jardinería, mediante un proceso ETL estructurado, ha permitido mejorar la calidad, organización y eficiencia en el manejo de los datos, proporcionando un entorno más adecuado para el análisis y la toma de decisiones.
- La implementación de un proceso ETL eficiente ha facilitado la extracción, transformación y carga de los datos, garantizando su limpieza, depuración y estructuración antes de ser cargados y almacenados en el Data Mart.
- La creación de una base de datos intermedia (Staging) ha permitido seleccionar y estructurar únicamente la información relevante, eliminando redundancias y asegurando la integridad de los datos.

- Se ha aplicado un proceso de normalización y asignación de claves foráneas, lo que garantiza la coherencia entre las tablas de hechos y dimensiones.
- La estructura del Data Mart permite realizar consultas más eficientes, reduciendo los tiempos de procesamiento y mejorando el rendimiento en la obtención de información clave para el negocio.
- Con la creación del Data Mart, se establece un punto de partida sólido para la implementación de sistemas de inteligencia de negocios (BI) y análisis avanzado de datos, lo que facilita la generación de reportes estratégicos.
- Las consultas SQL diseñadas para la migración de datos han permitido una transferencia eficiente, asegurando que la información se almacene correctamente sin pérdida de datos ni inconsistencias.
- La construcción del Data Mart ha demostrado ser una estrategia efectiva para mejorar la gestión y disponibilidad de la información, optimizando su uso en futuras consultas, reportes y análisis estratégicos dentro del sistema de Jardinería.

6. Anexos



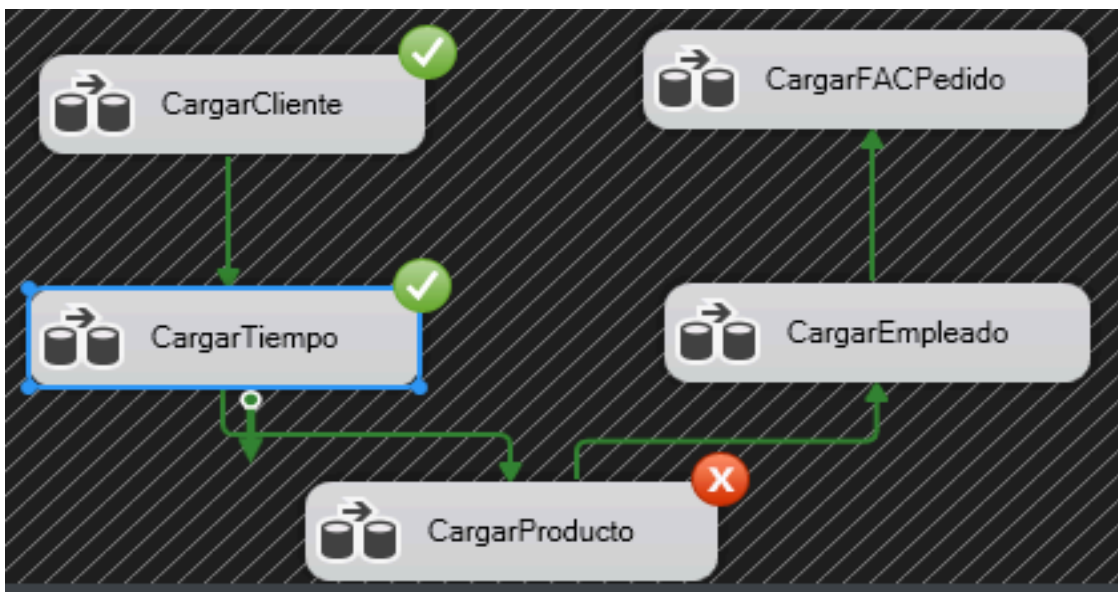
- Imagen del modelo estrella



- Imagen de la base de datos Staging (Extracción)



- Imagen proceso de Transformación



- Imagen proceso de Carga