

**Materia: Electiva II Bases de datos II**

**Docente: Victor Hugo Mercado**

## **Módulo 2**

**Creación de BD STAGING y  
Extracción de datos (ETL)**



**Grupo 5:**

- **Carlos Andrés Bustamante Montoya**
- **Santiago Osorio Henao**

**Especialización en Analítica y Big Data**

Medellín, 24 de Noviembre de 2024

# Diseño de un Modelo Estrella para el Análisis de Ventas en la Empresa "Jardinería"

## Introducción

El propósito de este proyecto es diseñar y crear una base de datos **Staging** a partir del modelo relacional de la base de datos **Jardinería**. La base de datos **Staging** actúa como una capa intermedia entre las fuentes de datos operacionales y el sistema de destino, donde se transforman, cargan y consolidan datos para su análisis posterior. Este enfoque es especialmente útil cuando se integran datos provenientes de diferentes fuentes y se necesitan preparar para la carga a un sistema de almacenamiento o análisis más complejo, como un Data Warehouse o Data Mark.

## Objetivo

- Diseñar y construir una base de datos **Staging** a partir de la base de datos **Jardinería**, optimizando la estructura y la transferencia de datos para futuros análisis e integraciones.
- Identificar los datos relevantes en la base de datos Jardinería que deben ser trasladados al entorno Staging.
- Construir la estructura adecuada de tablas en la base de datos Staging.
- Utilizar Visual Studio ETL para extraer, transformar y cargar (ETL) los datos relevantes desde Jardinería a Staging.
- Validar la correcta transferencia de los datos y crear respaldos de ambas bases de datos.

## Planteamiento del problema

La base de datos "Jardinería" actualmente contiene información detallada sobre clientes, empleados, productos, y transacciones, pero no está optimizada para realizar análisis de ventas de manera eficiente limitando su capacidad para obtener información detallada sobre las transacciones, productos, clientes y empleados involucrados. La implementación de un modelo en estrella permitirá una organización más intuitiva y eficiente de los datos, adecuada para consultas de análisis.

Los responsables de la toma de decisiones requieren una herramienta que les permita conocer de forma rápida y precisa los productos más vendidos, las categorías que generan más ingresos y el comportamiento de las ventas por año. Para resolver este problema, se necesita estructurar los datos de manera que se facilite el acceso a métricas clave a través de un modelo de datos adecuado.

## Análisis del problema

El principal desafío radica en la estructura actual de la base de datos, que almacena la información de forma relacional pero no está optimizada para consultas analíticas rápidas. Sin un modelo bien estructurado, los usuarios enfrentan dificultades para obtener respuestas rápidas sobre el desempeño de productos, categorías y periodos específicos.

El modelo de base de datos Jardinería contiene varias tablas interrelacionadas que manejan datos de clientes, empleados, productos, pedidos, pagos y más. Sin embargo, no todas las columnas de cada tabla son relevantes para el análisis en el entorno Staging. Por lo tanto, el análisis del problema consistió en seleccionar los campos más relevantes para ser trasladados a la nueva base de datos.

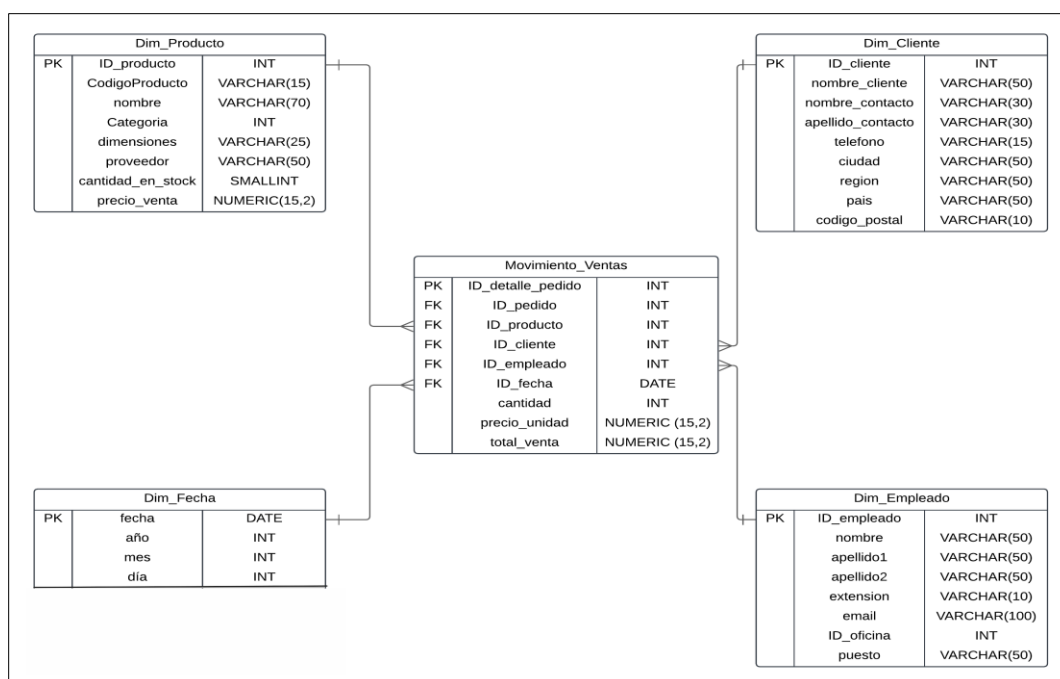
## Propuesta de solución

### Corrección Entrega 1:

### Descripción de modelo de estrella propuesto

El modelo estrella para "Jardinería" está centrado en la tabla de hechos llamada **"Movimiento\_Ventas"**, que representa cada transacción de venta de productos. Esta tabla se relaciona con dimensiones que describen los detalles contextuales de cada venta, como el producto, el cliente, el empleado que gestionó la venta y la fecha en que se realizó la transacción. Cada dimensión almacena detalles específicos que permiten analizar las ventas desde diferentes perspectivas.

### Diseño del modelo estrella



## **Lista de dimensiones propuestas**

### **1. Dimensión de Producto (dim\_producto)**

**Descripción:** Proporciona detalles de cada producto vendido.

**Columnas y tipos de datos:**

- **ID\_producto (INT)** - Llave primaria.
- **CodigoProducto (VARCHAR(15))** - Código único del producto.
- **nombre (VARCHAR(70))** - Nombre descriptivo del producto.
- **Categoria (INT)** - Identificador de la categoría del producto.
- **dimensiones (VARCHAR(25))** - Dimensiones del producto.
- **proveedor (VARCHAR(50))** - Nombre del proveedor del producto.
- **cantidad\_en\_stock (SMALLINT)** - Cantidad disponible en inventario.
- **precio\_venta (NUMERIC(15,2))** - Precio de venta del producto.

### **2. Dimensión de Cliente (dim\_cliente)**

**Descripción:** Contiene información sobre los clientes que realizan las compras.

**Columnas y tipos de datos:**

- **ID\_cliente (INT)** - Llave primaria.
- **nombre\_cliente (VARCHAR(50))** - Nombre de la empresa o cliente.
- **nombre\_contacto (VARCHAR(30))** - Nombre del contacto principal.
- **apellido\_contacto (VARCHAR(30))** - Apellido del contacto.
- **telefono (VARCHAR(15))** - Teléfono de contacto.
- **ciudad (VARCHAR(50))** - Ciudad de residencia.
- **region (VARCHAR(50))** - Región de residencia.
- **pais (VARCHAR(50))** - País de residencia.
- **codigo\_postal (VARCHAR(10))** - Código postal.

### **3. Dimensión de Empleado (dim\_empleado)**

**Descripción:** Registra información sobre los empleados que gestionan las ventas.

**Columnas y tipos de datos:**

- **ID\_empleado (INT)** - Llave primaria.
- **nombre (VARCHAR(50))** - Nombre del empleado.
- **apellido1 (VARCHAR(50))** - Primer apellido.
- **apellido2 (VARCHAR(50))** - Segundo apellido.
- **extension (VARCHAR(10))** - Extensión telefónica.
- **email (VARCHAR(100))** - Correo electrónico.

- **ID\_oficina (INT)** - Identificador de la oficina.
- **puesto (VARCHAR(50))** - Puesto del empleado.

#### 4. Dimensión de Fecha (dim\_fecha)

**Descripción:** Permite análisis mediante el desglose de la fecha de venta.

**Columnas y tipos de datos:**

- **fecha (DATE)** - Llave primaria.
- **año (INT)** - Año de la fecha.
- **mes (INT)** - Mes de la fecha.
- **día (INT)** - Día de la fecha.

#### Tabla Hechos

La tabla de hechos “**Movimiento\_Ventas**” almacena las transacciones de venta

- **ID\_detalle\_pedido (INT (PK))**- Identificador único de la transacción de venta.
- **ID\_pedido (INT)**- Relación con la tabla de pedidos.
- **ID\_producto (INT)**- Identificador del producto vendido.
- **ID\_cliente (INT)**- Identificador del cliente.
- **ID\_empleado (INT)**- Identificador del empleado de ventas.
- **ID\_fecha (DATE)**- Fecha de la venta.
- **Cantidad (INT)**- Cantidad de producto vendida.
- **precio\_unidad (NUMERIC(15,2))**- Precio de venta unitario.
- **total\_linea (NUMERIC(15,2))**- Total de la línea (cantidad \* precio\_unidad).

## Descripción del análisis y traslados de la información

El análisis se centró en los siguientes aspectos clave:

### Selección de Datos Relevantes:

Se eligieron los campos clave de cada tabla para garantizar que solo la información necesaria fuera extraída de la base de datos original Jardinería.

### Extracción de datos:

La extracción de datos se realizó utilizando Visual Studio ETL, específicamente a través de SQL Server Integration Services (SSIS). Este enfoque es ideal para proyectos de integración de datos, ya que permite mover, transformar y cargar grandes volúmenes de datos de manera eficiente.

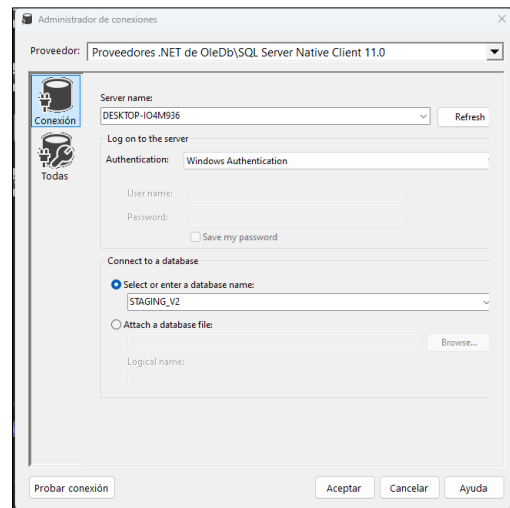
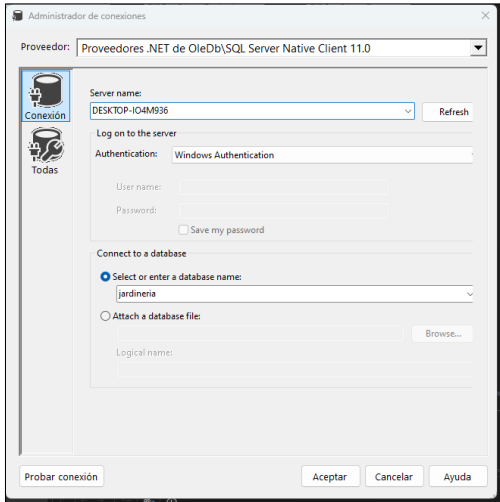
### Pasos de la Extracción:

#### 1-Creación del Proyecto ETL en Visual Studio:

Se comenzó creando un proyecto en Visual Studio a través de integration Services Project.

#### 2-Configuración de la Conexión:

Se configuró una conexión ADO.NET para acceder a la base de datos Jardinería (origen) y STAGIN\_V2 (Destino).

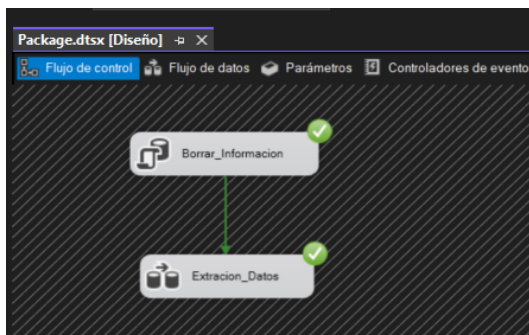
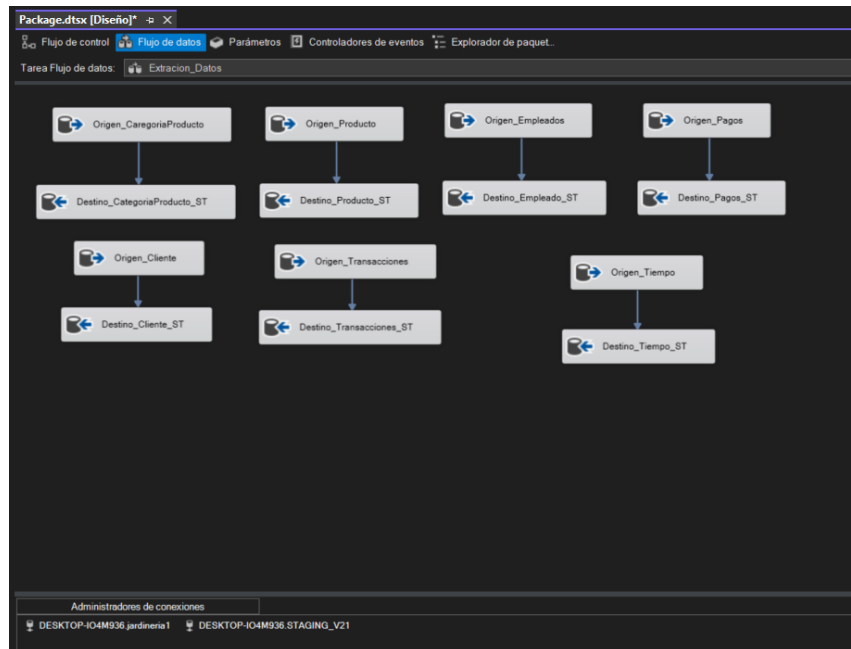


#### 3-Selección de Campos:

Es el resultado de la primera actividad donde después del análisis se mapean los campos relevantes de cada tabla de la base de datos Jardinería.

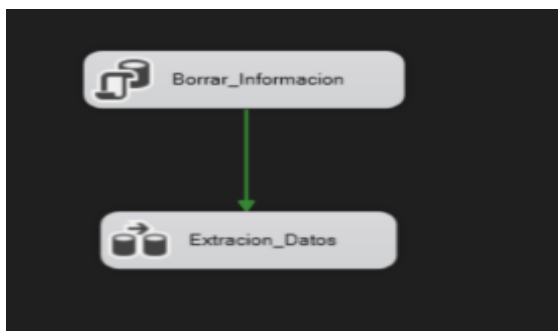
#### 4-Cargar los Datos en Staging:

Los datos fueron cargados en las tablas correspondientes dentro de la base de datos Staging, de la siguiente forma:



#### 5-Insertar Tarea SQL:

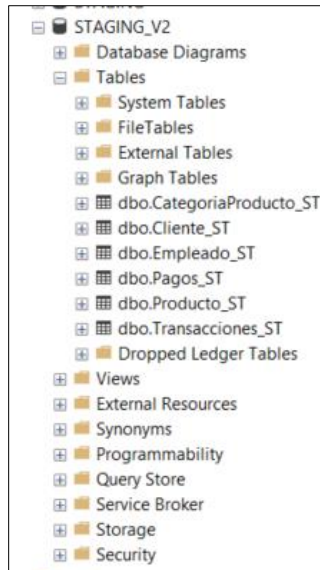
Se creó una tarea que ayuda a borrar la información antes de insertarla, quedando esta forma:



Primero se ejecuta la tarea que limpia la información de las tablas de STAGINGV2 y luego inserta los registros, y así evitar duplicidad en estos.

## 6-Validacion y estructura de la información:

Se verifica en la base de datos de destino la creación de las tablas y la inserción correcta de los datos:



Con esta información se tiene los campos necesarios para la creación del modelo relacional y seguir con el siguiente paso del ETL la transformación