



INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA

Base de Datos 2

S20 - Evidencia de aprendizaje 1. Modelo estrella de un Data Mart

Presentado Por:

Camilo Londoño Acebedo

Jerónimo Hernández Lopera

Jorge Luis Perez Severiche

Ingeniería de Software y Datos

Colombia

10 de Octubre de 2025

Introducción

En este trabajo se llevó a cabo la transformación de la base de datos transaccional **Jardinería** hacia un modelo analítico basado en un **esquema en estrella**, con el propósito de optimizar el análisis de la información y apoyar la toma de decisiones empresariales. Para lograrlo, se identificaron dos procesos de negocio principales: las **ventas** y los **pagos**, los cuales se representaron mediante las tablas de hechos **FactVentas** y **FactPagos**. Estas tablas se complementaron con dimensiones relevantes como cliente, producto, empleado, oficina, tiempo y forma de pago, que permiten examinar los datos desde múltiples perspectivas. De esta manera, el modelo diseñado integra la información de forma clara y estructurada.

Objetivos

Diseñar y construir un modelo de datos en estrella a partir de la base de datos transaccional **Jardinería**, con el fin de optimizar el análisis de información y mejorar la capacidad de respuesta en los procesos de toma de decisiones estratégicas.

Planteamiento del problema

La base de datos transaccional **Jardinería** fue diseñada para gestionar las operaciones diarias de la empresa, tales como la administración de pedidos, pagos, clientes, empleados y productos. Sin embargo, este tipo de modelo relacional no resulta eficiente para realizar análisis estratégicos, ya que su estructura está optimizada para registrar transacciones y no para responder preguntas de negocio de manera ágil.

Análisis del Problema

Revisa la estructura de la base de datos Jardinería para identificar las tablas relevantes y sus relaciones.

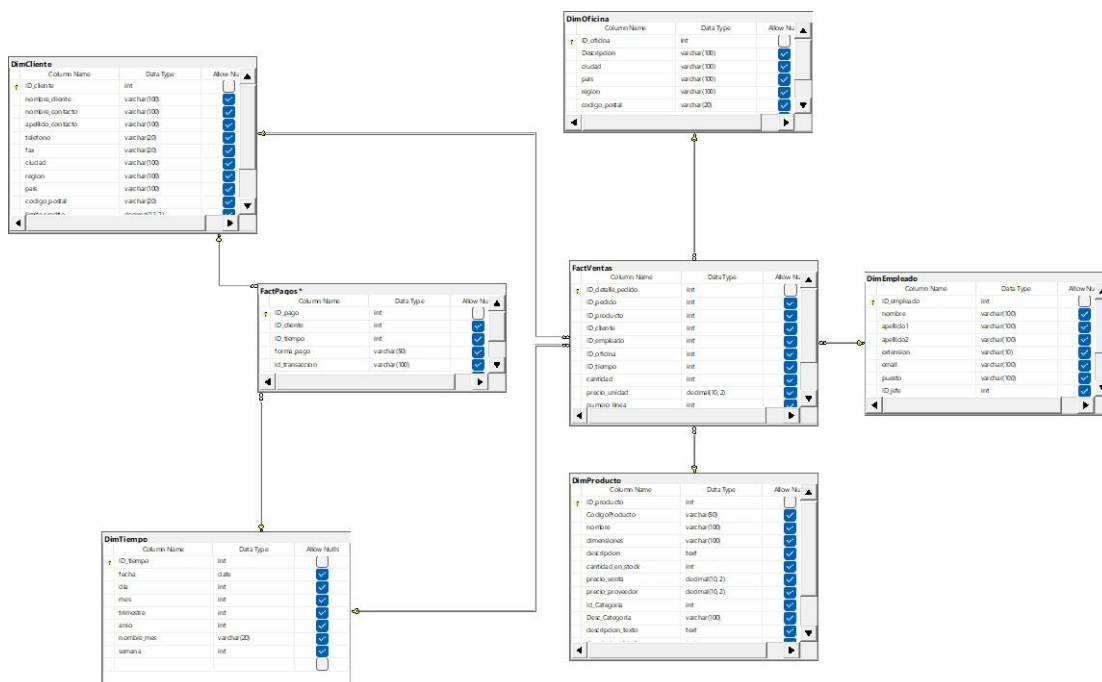
Después de ejecutar la Base de datos en el Motor SQL Server Management Studio y analizar las distintas tablas y sus relaciones descubrí que las tablas más importantes son Cliente y Pedido ya que son las que guardan la información más importante con las demás tablas y la manera en que se relacionan es la siguiente:

- Cliente → guarda los datos de los clientes. ○ Relación: tiene muchos Pedidos y Pagos, y está vinculado a un Empleado
- Pedido → almacena los pedidos realizados.
 - Relación: pertenece a un Cliente y se detalla en Detalle_pedido.

Propuesta de la solución:

Identifica los campos necesarios para construir el modelo estrella, incluyendo la tabla de hechos y las dimensiones pertinentes.

He decidido crear las tablas **FactVentas** y **FactPagos** porque en la base de datos Jardinería se identifican dos procesos de negocio independientes que requieren análisis diferenciados: las ventas, que corresponden a los pedidos realizados por los clientes e incluyen métricas como cantidades, precios unitarios y totales de línea; y los pagos, que reflejan las transacciones financieras realizadas por los clientes, las cuales no siempre coinciden en tiempo ni en monto con las ventas. Para la construcción del modelo en estrella, la tabla **FactVentas** incluye las claves de pedido, producto, cliente, empleado, oficina y fecha, mientras que la tabla **FactPagos** contiene las claves de cliente, fecha y forma de pago, junto con la métrica del monto pagado. En cuanto a las dimensiones, se incorporan **DimCliente** (información de los clientes), **DimProducto** (catálogo de productos y sus categorías), **DimEmpleado** (representantes de ventas y jerarquía), **DimOficina** (ubicación de oficinas), **DimTiempo** (fechas normalizadas para análisis temporal).



Diseña la estructura tabla de hechos que represente las ventas o transacciones de la empresa.

En la base de datos original esta la tabla pedidos y la tabla pagos como dos elementos diferenciados y de ahí me base para crear dos tablas **FactVentas** y **FactPagos**, como lo mencione anteriormente un cliente puede realizar un pedido de varios productos y puede realizar un pago que no necesariamente se corresponde con los pedidos realizados anteriormente por que en los comentarios podemos ver que se puede pagar a plazos y que un pago puede cubrir parcialmente un pedido o puede cubrir varios pedidos, básicamente hay dos transacciones realizadas por la empresa la primera venta cuando se realiza y entrega el pedido como una segunda transacción cuando el cliente paga.

FactVentas: Siendo esta la primera transacción realizada por la empresa donde se le entrega el producto a los clientes, decidí unificar aca varios de los IDs requeridos, ademas de agregar el precio total_venta que suma programáticamente la cantidad * precio_unidad.

FactVentas			
	Column Name	Data Type	Allow Nulls
🔑	ID_detalle_pedido	int	<input type="checkbox"/>
	ID_pedido	int	<input checked="" type="checkbox"/>
	ID_producto	int	<input checked="" type="checkbox"/>
	ID_cliente	int	<input checked="" type="checkbox"/>
	ID_empleado	int	<input checked="" type="checkbox"/>
	ID_oficina	int	<input checked="" type="checkbox"/>
	ID_tiempo	int	<input checked="" type="checkbox"/>
	cantidad	int	<input checked="" type="checkbox"/>
	precio_unidad	decimal(10, 2)	<input checked="" type="checkbox"/>
	numero_linea	int	<input checked="" type="checkbox"/>
	total_venta		<input checked="" type="checkbox"/>
			<input type="checkbox"/>

FactPagos: En esta tabla decido mantenerla casi igual a la de pago original de la base de datos jardinería, pero modificando la columna ID_tiempo, convirtiéndola de un tipo date a un tipo int para que se pueda relacionar con la tabla FactVentas

FactPagos			
	Column Name	Data Type	Allow Nulls
🔑	ID_pago	int	<input type="checkbox"/>
	ID_cliente	int	<input checked="" type="checkbox"/>
	ID_tiempo	int	<input checked="" type="checkbox"/>
	forma_pago	varchar(50)	<input checked="" type="checkbox"/>
	id_transaccion	varchar(100)	<input checked="" type="checkbox"/>
	total	decimal(12, 2)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Identifica y diseña las dimensiones relevantes que se relacionarán con la tabla de hechos.

DimCliente: Contiene la información de los clientes, como nombre, contacto y país, lo que permite analizar el comportamiento de compras y pagos según la localización

DimCliente			
	Column Name	Data Type	Allow Nulls
🔑	ID_cliente	int	<input type="checkbox"/>
	nombre_cliente	varchar(100)	<input checked="" type="checkbox"/>
	nombre_contacto	varchar(100)	<input checked="" type="checkbox"/>
	apellido_contacto	varchar(100)	<input checked="" type="checkbox"/>
	telefono	varchar(20)	<input checked="" type="checkbox"/>
	fax	varchar(20)	<input checked="" type="checkbox"/>
	ciudad	varchar(100)	<input checked="" type="checkbox"/>
	region	varchar(100)	<input checked="" type="checkbox"/>
	pais	varchar(100)	<input checked="" type="checkbox"/>
	codigo_postal	varchar(20)	<input checked="" type="checkbox"/>
	limite_credito	decimal(12, 2)	<input checked="" type="checkbox"/>
	ID_empleado_rep_ventas	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DimProducto: Almacena datos de los productos, incluyendo su categoría, línea y proveedor, facilitando el estudio de las ventas por tipo de producto.

DimProducto			
	Column Name	Data Type	Allow Nulls
🔑	ID_producto	int	<input type="checkbox"/>
	CodigoProducto	varchar(50)	<input checked="" type="checkbox"/>
	nombre	varchar(100)	<input checked="" type="checkbox"/>
	dimensiones	varchar(100)	<input checked="" type="checkbox"/>
	descripcion	text	<input checked="" type="checkbox"/>
	cantidad_en_stock	int	<input checked="" type="checkbox"/>
	precio_venta	decimal(10, 2)	<input checked="" type="checkbox"/>
	precio_proveedor	decimal(10, 2)	<input checked="" type="checkbox"/>
	Id_Categoria	int	<input checked="" type="checkbox"/>
	Desc_Categoria	varchar(100)	<input checked="" type="checkbox"/>
	descripcion_texto	text	<input checked="" type="checkbox"/>
	descripcion_html	text	<input checked="" type="checkbox"/>
	imagen	varchar(255)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DimEmpleado: Representa a los representantes de ventas y sus jefes, lo que posibilita medir el desempeño individual o por jerarquías de ventas

DimEmpleado			
	Column Name	Data Type	Allow Nulls
🔑	ID_empleado	int	<input type="checkbox"/>
	nombre	varchar(100)	<input checked="" type="checkbox"/>
	apellido1	varchar(100)	<input checked="" type="checkbox"/>
	apellido2	varchar(100)	<input checked="" type="checkbox"/>
	extension	varchar(10)	<input checked="" type="checkbox"/>
	email	varchar(100)	<input checked="" type="checkbox"/>
	puesto	varchar(100)	<input checked="" type="checkbox"/>
	ID_jefe	int	<input checked="" type="checkbox"/>
	ID_oficina	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DimOficina: Describe las oficinas en las que trabajan los empleados, aportando contexto geográfico y organizacional.

DimOficina			
	Column Name	Data Type	Allow Nulls
🔑	ID_oficina	int	<input type="checkbox"/>
	Descripcion	varchar(100)	<input checked="" type="checkbox"/>
	ciudad	varchar(100)	<input checked="" type="checkbox"/>
	pais	varchar(100)	<input checked="" type="checkbox"/>
	region	varchar(100)	<input checked="" type="checkbox"/>
	codigo_postal	varchar(20)	<input checked="" type="checkbox"/>
	telefono	varchar(20)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DimTiempo: Estructura las fechas en jerarquías de día, mes, trimestre y año, fundamentales para los análisis temporales de ventas y pagos.

DimTiempo			
	Column Name	Data Type	Allow Nulls
🔑	ID_tiempo	int	<input type="checkbox"/>
	fecha	date	<input checked="" type="checkbox"/>
	día	int	<input checked="" type="checkbox"/>
	mes	int	<input checked="" type="checkbox"/>
	trimestre	int	<input checked="" type="checkbox"/>
	año	int	<input checked="" type="checkbox"/>
	nombre_mes	varchar(20)	<input checked="" type="checkbox"/>
	semana	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Conclusiones

La transformación de la base de datos transaccional **Jardinería** en un modelo dimensional bajo un esquema en estrella permitió estructurar la información de manera más clara y eficiente para fines analíticos. La creación de las tablas de hechos **FactVentas** y **FactPagos**, junto con sus dimensiones asociadas, facilita la exploración de los datos desde múltiples perspectivas como clientes, productos, empleados, oficinas, tiempo y formas de pago. Este diseño reduce la complejidad de las consultas y mejora la capacidad de respuesta frente a preguntas de negocio estratégicas.

Bibliografia <https://learn.microsoft.com/en-us/answers/questions/2111481/star-schema-withmultiple-fact-tables>

Anexos

