

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

Integrantes

Didier Álvarez Bustamante

Jonathan Álvarez Bustamante

Institución Universitaria Digital de Antioquia
Facultad de Ingeniería y Ciencias Agropecuarias
Programa de Ingeniería de Software y Datos

Docente

Antonio Jesús Valderrama Jaramillo

PREICA2502B010064

Base de datos II

30 de septiembre, 2025

Tabla De Contenido

1. Introducción.....	3
2. Objetivos.....	3
3. Preparación.....	5
Archivos y bases de datos disponibles.	5
I. Base OLTP:	5
II. Base Staging:	5
III. Modelo Estrella (Data Mart):	5
Modelo estrella.	5
4. Extracción de datos.....	6
5. Transformación de los datos.....	7
Técnicas aplicadas	7
6. Carga de datos en el Data Mart.	8
7. Referencias.	8

1. Introducción.

En el presente informe se describe el proceso de **extracción, transformación y carga** (ETL) aplicado a la base de datos **Jardinería**.

El objetivo principal es disponer de un **Data Mart** con un **modelo estrella**, optimizado para la generación de reportes y análisis de ventas.

Este proceso ETL asegura que la información pase de un **sistema transaccional (OLTP)** a un **entorno analítico (OLAP)** mediante las etapas de extracción de los datos desde la base original, la transformación para asegurar calidad, coherencia y adecuación al modelo analítico y la carga en el modelo estrella (Data Mart) para su posterior explotación.

2. Objetivos.

General.

Desarrollar un flujo de integración de datos que abarque desde la base OLTP hasta la capa Staging y finalmente el modelo estrella. Este flujo debe garantizar la trazabilidad e integridad de la información al tiempo que documenta el diseño, las transformaciones aplicadas y los procesos de validación implementado para habilitar el análisis de ventas.

Específicos.

- Estandarizar la ingesta de datos en una capa de Staging que reciba la información del sistema OLTP sin imponer reglas de negocio estrictas, pero asegurando compatibilidad en los tipos de datos y la consistencia estructural.
- Diseñar un modelo estrella bien definido, con dimensiones y hecho organizados de manera que respondan eficientemente a las necesidades analíticas del negocio, permitiendo consultas por cliente, producto, representante de ventas, oficina, estado de pedido y periodos de tiempo.
- Estableces y aplicar reglas de transformación y carga que guíen el movimiento de los dato desde el OLTP hacia el Staging y desde allí hacia el Data Warehouse, mediante procesos ETL/ELT documentados y controlados.

3. Preparación.

Archivos y bases de datos disponibles.

I. Base OLTP:

- `DB_Jardineria.SQL` (estructura de tablas relacionales).
- Datos de `DB_Jardineria.SQL` (poblado inicial de datos).

II. Base Staging:

- `DB_Staging_Jardineria.SQL` (estructura de tablas intermedias).
- Datos de `DB_Staging_Jardineria.SQL` (migración de datos desde Jardinería).

III. Modelo Estrella (Data Mart):

- `DB_ModeloEstrella.SQL` (estructura de dimensiones y tabla de hechos).
- Datos de `DB_ModeloEstrella.SQL` (poblado de dimensiones y hechos).

Modelo estrella.

El modelo estrella diseñado contempla las siguientes tablas:

- **Dimensiones:** Cliente, Producto, Empleado, Oficina, Estado de Pedido, Tiempo.
- **Tabla de hechos:** `hechos_ventas`, que concentra las métricas de negocio (cantidad, precio, total de venta).

El diseño implementado permite consultas eficientes relacionadas con ventas por cliente, producto, oficina, estado del pedido o periodos de tiempo.

4. Extracción de datos.

La etapa de **extracción** consistió en trasladar los datos de la base transaccional (jardineria) hacia una base intermedia o **staging** (jardineria_staging).

Se utilizaron sentencias `INSERT INTO ... SELECT` que copian los registros desde las tablas originales hacia las tablas de staging, sin alterar la información.

Ejemplo de carga de clientes en Staging:

```
INSERT INTO jardineria_staging.dbo.Stg_cliente (ID_cliente, nombre_cliente, ciudad, region,
pais, ID_empleado_rep_ventas)

SELECT ID_cliente, nombre_cliente, ciudad, region, pais, ID_empleado_rep_ventas

FROM jardineria.dbo.cliente;
```

De esta forma se garantizó que toda la información de clientes, empleados, oficinas, pedidos, detalles de pedido, productos, categorías y pagos quedara disponible en staging, sin afectar la base de datos original.

5. Transformación de los datos.

En esta fase se prepararon los datos para que cumplieran con los **requisitos analíticos** del modelo estrella.

Técnicas aplicadas

I. Limpieza:

- Uso de `DISTINCT` para eliminar registros duplicados en dimensiones como cliente, producto u oficina.
- Eliminación de nulos en campos obligatorios (ej. fechas en pedidos).

II. Normalización:

- Establecimiento de claves uniformes para todas las dimensiones.
- Creación de la dimensión tiempo con claves en formato `YYYYMMDD` para facilitar análisis por año, mes y día.

III. Enriquecimiento de datos:

- Uso de `JOIN` para complementar información.
- Ejemplo: unir `producto` con `categoría` para que la dimensión de producto incluya no solo el ID de la categoría, sino también su descripción textual.

Ejemplo de transformación de productos y categorías:

```
INSERT INTO dim_producto (id_producto, nombre_producto, categoria, desc_categoria, proveedor,
precio_venta)

SELECT DISTINCT p.ID_producto, p.nombre, p.ID_Categoria, c.Desc_Categoria, p.proveedor,
p.precio_venta

FROM jardineria_staging.dbo.Stg_producto p

JOIN jardineria_staging.dbo.Stg_Categoria_producto c

ON p.ID_Categoria = c.Id_Categoria;
```

De esta forma, cada dimensión contiene información única, coherente y enriquecida.

6. Carga de datos en el Data Mart.

La fase de **carga** consistió en insertar los registros ya transformados en las tablas del modelo estrella (`jardineria_dw`).

El procedimiento seguido fue:

- Poblar las dimensiones primero (cliente, producto, empleado, oficina, estado de pedido, tiempo).
- Poblar después la tabla de hechos `hechos_ventas`, asegurando la integridad referencial con las dimensiones.

Ejemplo de carga de la tabla de hechos:

```
INSERT INTO hechos_ventas (id_fecha, id_cliente, id_producto, id_empleado, id_oficina, id_estado,
cantidad, precio_unitario, total_venta)

SELECT

CAST(CONVERT(VARCHAR, p.fecha_pedido, 112) AS INT) AS id_fecha,
p.ID_cliente,
dp.ID_producto,
c.ID_empleado_rep_ventas,
o.ID_oficina,
ep.id_estado,
dp.cantidad,
dp.precio_unidad,
dp.cantidad * dp.precio_unidad AS total_venta
FROM jardineria_staging.dbo.Stg_detalle_pedido dp
JOIN jardineria_staging.dbo.Stg_pedido p ON dp.ID_pedido = p.ID_pedido
JOIN jardineria_staging.dbo.Stg_cliente c ON p.ID_cliente = c.ID_cliente
JOIN jardineria_staging.dbo.Stg_empleado e ON c.ID_empleado_rep_ventas = e.ID_empleado
JOIN jardineria_staging.dbo.Stg_oficina o ON e.ID_oficina = o.ID_oficina
JOIN dim_estado_pedido ep ON p.estado = ep.estado
```

7. Referencias.

- Golfarelli, M., & Rizzi, S. (2009). Data warehouse design: Modern principles and methodologies. McGraw-Hill.

- Hernández, J., & Muñoz, J. (2019). Bases de datos: Diseño y aplicaciones (2ª ed.). Alfaomega.
- Inmon, W. H. (2005). Building the data warehouse (4th ed.). Wiley.
- Kimball, R., & Caserta, J. (2004). The data warehouse ETL toolkit: Practical techniques for extracting, cleaning, conforming, and delivering data. Wiley.
- Kimball, R., & Ross, M. (2013). The data warehouse toolkit: The definitive guide to dimensional modeling (3rd ed.). Wiley.
- Microsoft. (2022). SQL Server Integration Services (SSIS) documentation. Microsoft. <https://learn.microsoft.com/sql/integration-services/>
- Oracle Corporation. (2023). Oracle data warehousing guide. Oracle. <https://docs.oracle.com/>
- Ponniah, P. (2010). Data warehousing fundamentals for IT professionals (2nd ed.). Wiley.