

EVIDENCIA DE APRENDIZAJE 2



CREACIÓN DE UNA BASE DE DATOS DE STAGING

PRESENTADO POR:
CARLOS ESTEBAN TRUJILLO PAZ
JOSÉ DAVID CARDONA MAZO
DANIELA LOZANO MORALES
JUAN FELIPE PARRA PALACIOS
DIEGO BARROS

INGENIERÍA DE SOFTWARE Y DATOS
BASES DE DATOS II
PREICA2502B010064

DOCENTE:
ANTONIO JESÚS VALDERRAMA JARAMILLO

INSTITUCION UNIVERSITARIA DIGITAL DE ANTIOQUIA
MEDELLIN, COLOMBIA

SEPTIEMBRE 2025

Resumen

En el escenario empresarial actual, el eficiente procesamiento de inmensas cantidades de datos es esencial para tomar decisiones de nivel de estrategia. Este proyecto abarca el desarrollo de una base de datos de zona intermedia Staging entre una base de datos transaccional denominada "Jardinería" y una base de datos dimensional del tipo estrella con el propósito de transformar datos brutos en datos útiles y jerarquizados.

El problema principal surge de la alta normalización de la base Jardinería, que aunque es adecuada para operaciones transaccionales, genera dificultades para el análisis debido a la fragmentación de información, datos redundantes e inconsistentes, y la necesidad de múltiples uniones complejas entre tablas. Esta estructura impacta negativamente el rendimiento cuando se realizan consultas analíticas.

La solución implementada consiste en una base Staging con menor normalización que facilita la limpieza y reorganización de datos. Se aplicaron mejoras significativas como la implementación de procesos ETL completos, reorganización de dimensiones eliminando redundancias, y fortalecimiento de la integridad referencial entre tablas.

El proceso de migración sigue la metodología ETL en tres fases: extracción mediante consultas SQL especializadas, transformación para limpieza y ajuste de formatos, y carga de datos procesados. El proyecto se enfoca especialmente en la fase de extracción, considerándola fundamental para el éxito del sistema completo y la efectividad del análisis posterior de la información empresarial.

Palabras clave: Staging, ETL (extracción, transformación y carga), integridad referencial, optimización de datos, dimensiones y hechos, normalización y desnormalización.

1. Introducción

En la actualidad, las organizaciones producen grandes volúmenes de datos y su correcto procesamiento resulta esencial para la toma de decisiones. Para evitar errores, duplicaciones o pérdidas de información, es necesario contar con procedimientos que transforman los datos en bruto en información valiosa.

Este informe trata sobre la creación de una base de datos Staging, que funciona como una zona intermedia entre la base transaccional Jardinería y el modelo dimensional (modelo estrella). En esta fase se realiza el análisis de la información, los procesos de extracción y carga, así como las adaptaciones necesarias para asegurar la calidad de los datos antes de trasladarlos al modelo final.

2. Objetivos

- Diseñar e implementar una base de datos Staging que sirva como puente de integración entre Jardinería y el modelo estrella.
- Desarrollar y ejecutar consultas SQL para trasladar la información de forma ordenada y sin perder integridad.

- Documentar y validar las correcciones aplicadas en comparación con la entrega anterior, asegurando que los datos estén correctamente estructurados

3. Planteamiento del Problema

La base de datos Jardinería está muy normalizada. Aunque este diseño resulta útil para las operaciones transaccionales, plantea complicaciones cuando se quieren hacer consultas de análisis, ya que requiere muchas uniones y puede afectar el rendimiento.

El principal problema es que no se puede usar directamente la base transaccional para analizar, porque esto podría sobrecargar el sistema y ralentizar las operaciones. La solución más adecuada es crear una base de datos Staging, en la que los datos se depuren, se reorganicen y queden listos para cargarse en el modelo estrella.

4. Análisis del problema

- Durante el análisis de Jardinería se detectaron varios retos: Fragmentación de información: los datos están repartidos en muchas tablas.

- Datos redundantes o inconsistentes: es necesario limpiarlos antes de usarlos.

- Uniones complejas: para obtener datos consolidados hay que hacer muchas relaciones.

Para solucionar esto, se diseñó una base de datos Staging con menos normalización, que almacena la información en un formato más práctico para integrar en el modelo dimensional.

5. Ajustes realizados de la entrega 1

Se hicieron las siguientes mejoras:

- Se añadió un proceso ETL (Extracción, Transformación y Carga) para preparar los datos.

- Se ajustaron las dimensiones, eliminando redundancias y reorganizando las claves.

- Se reforzó la integridad referencial, asegurando que las relaciones entre tablas sean coherentes.

6. Proceso de migración de datos

El traslado de la información se realizó en tres fases:

- Extracción: obtención de datos relevantes mediante consultas SQL.

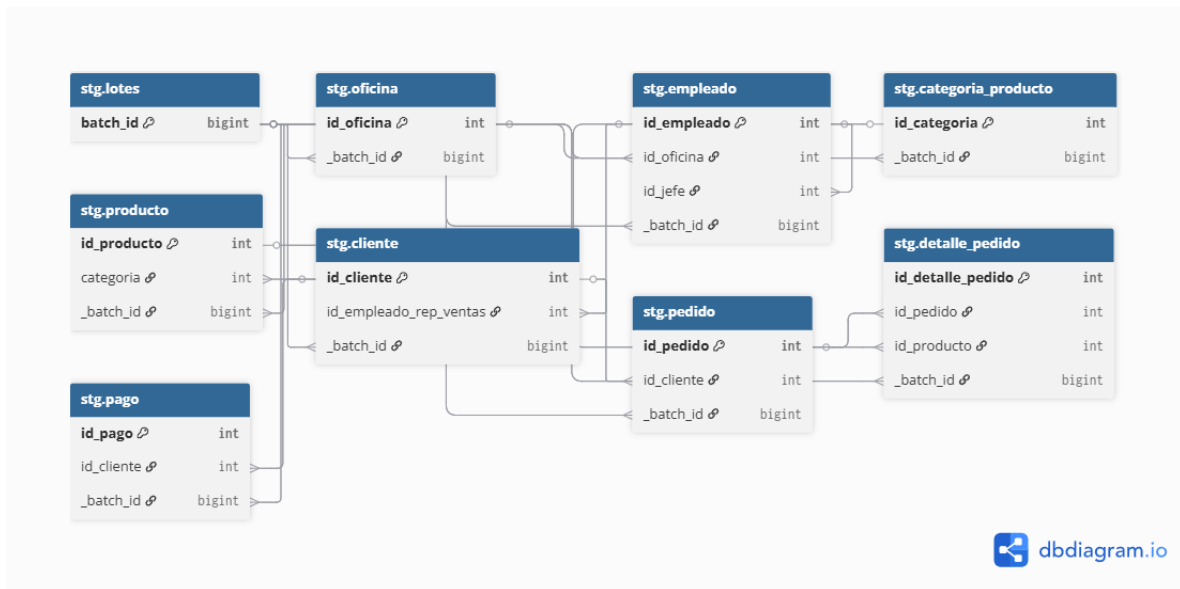
- Transformación: limpieza de registros, ajustes de formato y corrección de inconsistencias.

- Carga: inserción de los datos transformados en la base Staging para su posterior análisis.

En este trabajo, el énfasis se encuentra en la primera fase (extracción), ya que es la base para que las otras etapas funcionen correctamente.

Es importante aclarar que en esta base de datos hemos creado una tabla de auditoría / control para: Controlar las cargas, Auditar el origen y la calidad, Relacionar con las tablas de staging, Facilitar re-procesos

7. Estructura de la Base de Datos Staging



8. Arquitectura del Modelo Estrella

----- Auditoria

stg.lotes (

```

batch_id    BIGINT IDENTITY(1,1) PRIMARY KEY,
source_system VARCHAR(50) NOT NULL DEFAULT('JARDINERIA'),
started_at  DATETIME2(0) NOT NULL DEFAULT(SYSDATETIME()),
ended_at    DATETIME2(0) NULL,
status      VARCHAR(20) NOT NULL DEFAULT('RUNNING'),
rows_loaded INT      NOT NULL DEFAULT(0),

```

```

    notes      VARCHAR(4000) NULL
);

```

--- Tablas Maestras

```

stg.oficina (
    id_oficina    INT      NULL,
    descripcion    VARCHAR(10) NULL,
    ciudad        VARCHAR(30) NULL,
    pais          VARCHAR(50) NULL,
    region        VARCHAR(50) NULL,
    codigo_postal  VARCHAR(10) NULL,
    telefono      VARCHAR(20) NULL,
    linea_direccion1 VARCHAR(50) NULL,
    linea_direccion2 VARCHAR(50) NULL,
    _batch_id     BIGINT   NOT NULL,
    _extract_ts   DATETIME2(0) NOT NULL,
    _source       VARCHAR(50) NOT NULL DEFAULT('jardineria.oficina')
);

```

```

stg.empleado (
    id_empleado    INT      NULL,
    nombre         VARCHAR(50) NULL,

```



```

apellido1    VARCHAR(50) NULL,
apellido2    VARCHAR(50) NULL,
extension    VARCHAR(10) NULL,
email        VARCHAR(100) NULL,
id_oficina   INT      NULL,
id_jefe      INT      NULL,
puesto       VARCHAR(50) NULL,
_batch_id    BIGINT   NOT NULL,
_extract_ts  DATETIME2(0) NOT NULL,
_source      VARCHAR(50) NOT NULL DEFAULT('jardineria.empleado')
);

```

```

stg.categoria_producto (
    id_categoria    INT      NULL,
    desc_categoria  VARCHAR(50) NULL,
    descripcion_texto VARCHAR(MAX) NULL,
    descripcion_html VARCHAR(MAX) NULL,
    imagen          VARCHAR(256) NULL,
    _batch_id       BIGINT   NOT NULL,
    _extract_ts     DATETIME2(0) NOT NULL,
    _source          VARCHAR(50)      NOT NULL
    DEFAULT('jardineria.categoria_producto')
);

```

```
stg.producto (  
    id_producto    INT        NULL,  
    codigo_producto VARCHAR(15) NULL,  
    nombre         VARCHAR(70) NULL,  
    categoria      INT        NULL,  
    dimensiones    VARCHAR(25) NULL,  
    proveedor      VARCHAR(50) NULL,  
    descripcion     VARCHAR(MAX) NULL,  
    cantidad_en_stock SMALLINT  NULL,  
    precio_venta    DECIMAL(15,2) NULL,  
    precio_proveedor DECIMAL(15,2) NULL,  
    _batch_id      BIGINT      NOT NULL,  
    _extract_ts     DATETIME2(0) NOT NULL,  
    _source         VARCHAR(50) NOT NULL DEFAULT('jardineria.producto')  
);
```

```
stg.cliente (  
    id_cliente      INT        NULL,  
    nombre_cliente  VARCHAR(50) NULL,  
    nombre_contacto VARCHAR(30) NULL,  
    apellido_contacto VARCHAR(30) NULL,  
    telefono        VARCHAR(15) NULL,
```

```

fax          VARCHAR(15) NULL,
linea_direccion1  VARCHAR(50) NULL,
linea_direccion2  VARCHAR(50) NULL,
ciudad        VARCHAR(50) NULL,
region        VARCHAR(50) NULL,
pais          VARCHAR(50) NULL,
codigo_postal   VARCHAR(10) NULL,
id_empleado_rep_ventas INT      NULL,
limite_credito  DECIMAL(15,2) NULL,
_batch_id      BIGINT      NOT NULL,
_extract_ts    DATETIME2(0) NOT NULL,
_source        VARCHAR(50) NOT NULL DEFAULT('jardineria.cliente')
);

```

---- Tablas Transaccionales

```

stg.pedido (
  id_pedido    INT      NULL,
  fecha_pedido DATE      NULL,
  fecha_esperada DATE     NULL,
  fecha_entrega DATE      NULL,
  estado       VARCHAR(15) NULL,
  comentarios  VARCHAR(MAX) NULL,
  id_cliente   INT      NULL,

```

```

_batch_id    BIGINT    NOT NULL,
_extract_ts  DATETIME2(0) NOT NULL,
_source      VARCHAR(50) NOT NULL DEFAULT('jardineria.pedido')
);

```

```

stg.detalle_pedido (
    id_detalle_pedido INT        NULL,
    id_pedido         INT        NULL,
    id_producto       INT        NULL,
    cantidad          INT        NULL,
    precio_unidad     DECIMAL(15,2) NULL,
    numero_linea      SMALLINT    NULL,
    _batch_id         BIGINT      NOT NULL,
    _extract_ts       DATETIME2(0) NOT NULL,
    _source            VARCHAR(50)      NOT NULL
    DEFAULT('jardineria.detalle_pedido')
);

```

```

stg.pago (
    id_pago    INT        NULL,
    id_cliente INT        NULL,
    forma_pago VARCHAR(40) NULL,

```

```

id_transaccion VARCHAR(50) NULL,
fecha_pago DATE NULL,
total DECIMAL(15,2) NULL,
_batch_id BIGINT NOT NULL,
_extract_ts DATETIME2(0) NOT NULL,
_source VARCHAR(50) NOT NULL DEFAULT('jardineria.pago')
);

```

9. Conclusión

En nuestra base de datos de Staging buscamos transformar un entorno transaccional en un espacio preparado para el análisis de datos. Los datos alojados en la base jardinería fueron depurados, reorganizados y listos para cargarse en el modelo estrella, lo que a futuro esperamos nos facilite su aprovechamiento en herramientas de inteligencia de negocios.

Agregar en este proyecto el proceso ETL fue clave, ya que nos permitió asegurar la calidad de los datos mediante una extracción planificada, la transformación para unificar formatos y corregir inconsistencias que se pudieran presentar en los datos originales, adicionalmente de una carga eficiente en las tablas intermedias. Con esto garantizamos que la información que llegará al modelo dimensional sea confiable, coherente y apta para consultas avanzadas sin comprometer el rendimiento de la base original.

Queremos adicionalmente agregar que trabajar con lotes o tabla de lotes nos permitió tener una idea más clara de organización en nuestra base de datos donde

comprendimos que así podemos tener más control sobre el flujo de cargas que se generen a través del tiempo en nuestra base de datos de Staging.

Referencias

Amazon Web Services, Inc. (n.d.). ¿Qué es ETL? – Explicación de extracción, transformación y carga (ETL). Recuperado de: <https://aws.amazon.com/es/what-is/etl/>

IBM. (n.d.). Qué es el proceso de extracción y transformación de carga. Recuperado de: <https://www.ibm.com/eses/topics/etl>