Proyecto Fase 1

Seminario de Sistemas 2 Sección A

Kevin Steve Martinez Lemus
202004816 | USAC

ÍNDICE

INTRODUCCIÓN

El presente manual técnico proporciona una guía detallada sobre el proceso de extracción, transformación y carga de datos (ETL) utilizando SQL Server Integration Services (SSIS). A lo largo de este documento, se describirán los pasos necesarios para realizar la integración de datos desde múltiples fuentes, su procesamiento y carga en bases de datos relacionales.

PROCESO ETL

Extraction

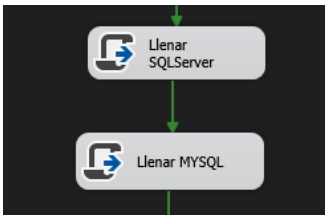
En esta sección, se detalla el proceso de extracción de datos realizado como parte del flujo de trabajo en el proyecto. La extracción se lleva a cabo principalmente desde archivos de texto plano con extensiones .comp y .vent, que contienen información relevante sobre compras y ventas, respectivamente.

Archivos de Texto Plano:

- Los archivos de texto plano .comp contienen datos relacionados con las compras.
- Los archivos de texto plano .vent contienen datos relacionados con las ventas.
- Estos archivos se someten a un proceso de lectura y análisis para extraer la información necesaria antes de ser cargados en el sistema.

Carga en Bases de Datos:

- Un archivo de ventas y un archivo de compras se cargan en una base de datos SQL Server.
- Un archivo de ventas y un archivo de compras se cargan en una base de datos MySQL.



Esto se realiza con el siguiente código el cual lee el archivo en cuestión por medio de un script y lo inserta en las tablas temporales de sus diferentes db:

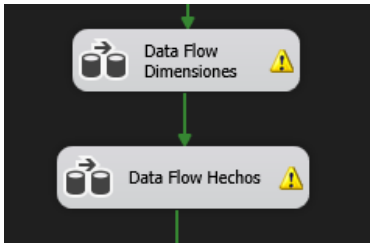
```
System.IO.StreamReader sourceFile = new System.IO.StreamReader(archivoCompra);


while((linea = sourceFile.ReadLine()) != null) {
    if (contador > 0)
    {
        string[] campos = linea.Split(delimitador.ToCharArray()[0]);

        string query = "INSERT INTO " + tablaCompra + " (fecha, codproveedor, nombreproveedor, direccionproveedor, numeroproveedor, webprov
            " (@Fecha, @Codproveedor, @Nombreproveedor, @Direccionproveedor, @Numeroproveedor, @Webproveedor, @Codproducto, @Nombreproducto

        using (SqlCommand myCommand = new SqlCommand(query, myADONETConnection))
        {
            myCommand.Parameters.AddWithValue("@Fecha", campos.Length > 0 ? campos[0] : "");
            myCommand.Parameters.AddWithValue("@Codproveedor", campos.Length > 1 ? campos[1] : "");
            myCommand.Parameters.AddWithValue("@Nombreproveedor", campos.Length > 2 ? campos[2] : "");
            myCommand.Parameters.AddWithValue("@Direccionproveedor", campos.Length > 3 ? campos[3] : "");
            myCommand.Parameters.AddWithValue("@Numeroproveedor", campos.Length > 4 ? campos[4] : "");
            myCommand.Parameters.AddWithValue("@Webproveedor", campos.Length > 5 ? campos[5] : "");
            myCommand.Parameters.AddWithValue("@Codproducto", campos.Length > 6 ? campos[6] : "");
            myCommand.Parameters.AddWithValue("@Nombreproducto", campos.Length > 7 ? campos[7] : "");
            myCommand.Parameters.AddWithValue("@Marcaproducto", campos.Length > 8 ? campos[8] : "");
            myCommand.Parameters.AddWithValue("@Categoria", campos.Length > 9 ? campos[9] : "");
            myCommand.Parameters.AddWithValue("@Sodsucursal", campos.Length > 10 ? campos[10] : "");
            myCommand.Parameters.AddWithValue("@Nombresucursal", campos.Length > 11 ? campos[11] : "");
            myCommand.Parameters.AddWithValue("@Direccionsucursal", campos.Length > 12 ? campos[12] : "");
            myCommand.Parameters.AddWithValue("@Region", campos.Length > 13 ? campos[13] : "");
            myCommand.Parameters.AddWithValue("@Departamento", campos.Length > 14 ? campos[14] : "");
            myCommand.Parameters.AddWithValue("@Unidades", campos.Length > 15 ? campos[15] : "");
            myCommand.Parameters.AddWithValue("@Costou", campos.Length > 16 ? campos[16] : "");

            myCommand.ExecuteNonQuery();
        }

    }
    contador++;
}
```

Extracción Directa desde Archivo:

- Además de la extracción desde archivos de texto plano, se realiza la extracción directa de datos desde un archivo específico.
- Este proceso implica la lectura directa del archivo sin procesamiento previo, lo que permite la obtención rápida de la información contenida en dicho archivo.

Transform

En esta sección, se describe el proceso de transformación de datos llevado a cabo como parte del flujo de trabajo en el proyecto. La transformación se centra principalmente en la conversión de tipos de datos, como la conversión de números a enteros, la cantidad de productos a enteros y el precio unitario a cadena de texto. Además, se ha realizado la eliminación de códigos nulos para mejorar la calidad de los datos.

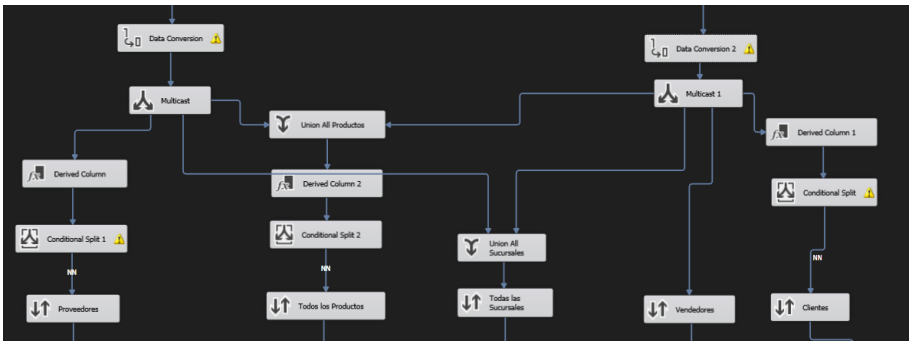


Conversión de Tipos de Datos:

- Se ha realizado la conversión de números a enteros para aquellos campos que representan valores numéricos enteros, como la cantidad de productos.
- La cantidad de productos se ha convertido de tipo de datos de cadena de texto a tipo de datos entero para facilitar el análisis y la manipulación de los datos.
- El precio unitario se ha convertido de tipo de datos de punto flotante a cadena de texto para preservar la precisión y evitar problemas de redondeo en cálculos futuros.

Eliminación de Códigos Nulos:

- Se han eliminado los códigos nulos o valores nulos de los datos para mejorar la integridad y consistencia de los mismos.
- Esta acción se ha llevado a cabo para garantizar que los datos sean válidos y estén completos antes de continuar con el procesamiento y análisis adicionales.

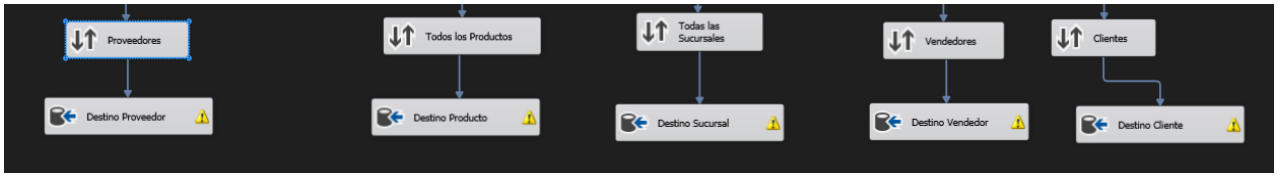


Load

En esta sección, se detalla el proceso de carga de datos una vez que han sido transformados y están listos para ser almacenados en sus respectivas tablas. Esto incluye la carga de datos en las tablas de dimensiones, que contienen información detallada sobre los elementos principales del negocio, así como en las tablas de hechos, que contienen datos transaccionales clave.

Carga en Tablas de Dimensiones:

- Los datos transformados se han cargado en las tablas de dimensiones correspondientes, que almacenan información sobre entidades clave como clientes, proveedores y productos.
- Cada tabla de dimensiones ha sido actualizada con los datos transformados y limpios, lo que garantiza la integridad y la calidad de los datos en todo momento.



Carga en Tablas de Hechos:

- Los datos transformados se han cargado en las tablas de hechos, que contienen información transaccional detallada, como ventas y compras.
- Cada fila de datos en las tablas de hechos representa una transacción individual y contiene referencias a las dimensiones pertinentes, lo que permite un análisis detallado y granular de las actividades comerciales.
- Se han aplicado restricciones de integridad referencial para garantizar que los datos cargados cumplan con las reglas de negocio y mantengan la coherencia entre las tablas de dimensiones y las tablas de hechos.

MODELO IMPLEMENTADO

Se utilizaron las estructuras de las tablas VentaTemp y CompraTemp, que sirven como punto de partida para el almacenamiento de datos durante el proceso de extracción y transformación. Estas tablas contienen todos los campos de los archivos de texto en formato VARCHAR, lo que proporciona flexibilidad para la carga inicial de datos sin restricciones de tipo de datos específicos. Esta decisión se tomó para permitir una mayor agilidad en el proceso de carga inicial y facilitar la posterior transformación de los datos según sea necesario para ajustarse al modelo de datos final.

```sql
CREATE TABLE CompraTemp (
    fecha varchar(50),
    codProveedor varchar(50),
    nombreProveedor varchar(100),
    direccionProveedor varchar(150),
    numeroProveedor varchar(50),
    webProveedor varchar(100),
    codProducto varchar(50),
    nombreProducto varchar(100),
    marcaProducto varchar(100),
    categoria varchar(100),
    sodSucursal varchar(50),
    nombreSucursal varchar(100),
    direccionSucursal varchar(150),
    region varchar(50),
    departamento varchar(50),
    unidades varchar(50),
    costoU varchar(50)
);
```

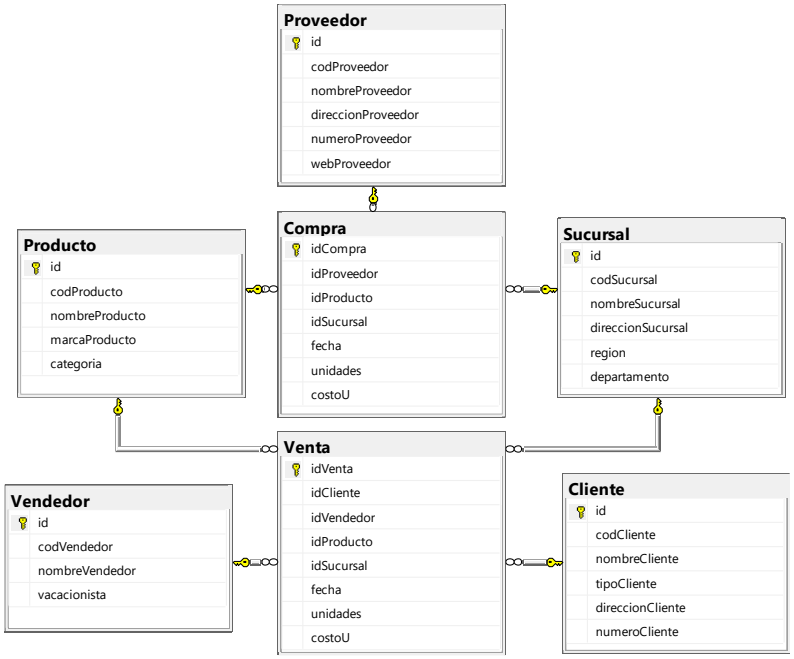
```sql
CREATE TABLE ventaTemp (
    fecha varchar(50),
    codCliente varchar(50),
    nombreCliente varchar(100),
    tipoCliente varchar(50),
    direccionCliente varchar(150),
    numeroCliente varchar(50),
    codVendedor varchar(50),
    nombreVendedor varchar(100),
    vacacionista varchar(50),
    codProducto varchar(50),
    nombreProducto varchar(100),
    marcaProducto varchar(100),
    categoria varchar(100),
    sodSucursal varchar(50),
    nombreSucursal varchar(100),
    direccionSucursal varchar(150),
    region varchar(50),
    departamento varchar(50),
    unidades varchar(50),
    precioUnitario varchar(50)
);
```

Para el diseño del data warehouse, se optó por utilizar el modelo estrella, una estructura de modelado dimensional que consta de una tabla central de hechos rodeada por tablas de dimensiones. En este modelo, la tabla central de hechos representa los eventos de negocio que se registran y se analizan, mientras que las tablas de dimensiones contienen atributos descriptivos que proporcionan contexto a estos eventos.

En el contexto de este proyecto, las tablas de hechos son Compra y Venta, que registran transacciones de compra y venta respectivamente. Estas tablas contienen las métricas principales de interés, como unidades vendidas, costo unitario, y otras medidas relevantes para el análisis del negocio.

Por otro lado, las tablas de dimensiones incluyen entidades como Proveedor, Sucursal, Cliente, Vendedor y Producto, que proporcionan detalles adicionales y contextuales sobre los eventos de compra y venta. Estas dimensiones contienen atributos descriptivos que ayudan a analizar y entender mejor los datos de las transacciones.

El modelo estrella facilita consultas analíticas complejas y eficientes al separar los datos en dimensiones y hechos, permitiendo un fácil acceso y análisis de la información empresarial.



Tablas de Hechos

Compra:

- Esta tabla representa las transacciones de compra realizadas en el sistema.
- Contiene campos como idCompra, fecha, unidades y costoU, que describen cada compra en detalle.
- Se relaciona con las tablas de dimensiones de Proveedor, Sucursal y Producto a través de llaves foráneas.
- La llave primaria de esta tabla es idCompra, que sirve como identificador único para cada registro.

Venta:

- Esta tabla registra las transacciones de venta realizadas en el sistema.
- Incluye campos como idVenta, fecha, unidades y costoU, que capturan información relevante sobre cada venta.
- Está relacionada con las tablas de dimensiones de Producto, Sucursal, Vendedor y Cliente mediante llaves foráneas.

- La llave primaria de esta tabla es idVenta, que identifica de manera única cada registro de venta.

Tablas de Dimensiones

1. **Proveedor:**
   - Esta tabla almacena información sobre los proveedores involucrados en las transacciones de compra.
   - Contiene atributos como idProveedor, nombreProveedor y otros detalles relacionados con cada proveedor.
   - Se utiliza como una dimensión en el contexto de las transacciones de compra.
2. **Sucursal:**
   - Representa las sucursales o puntos de venta asociados con las transacciones de compra y venta.
   - Incluye campos como idSucursal, nombreSucursal, dirección, etc., para describir cada sucursal de manera única.
   - Se relaciona con las tablas de hechos de Compra y Venta a través de llaves foráneas.
3. **Cliente:**
   - Almacena información sobre los clientes que participan en las transacciones de venta.
   - Contiene atributos como idCliente, nombreCliente, tipo de cliente, etc., para describir cada cliente.
   - Se utiliza como dimensión en el contexto de las transacciones de venta.
4. **Vendedor:**
   - Esta tabla registra detalles sobre los vendedores involucrados en las transacciones de venta.
   - Incluye campos como idVendedor, nombreVendedor, vacacionista, etc., para describir a cada vendedor de manera única.
   - Se utiliza como dimensión en el contexto de las transacciones de venta.
5. **Producto:**
   - Almacena información sobre los productos involucrados en las transacciones de compra y venta.
   - Contiene atributos como idProducto, nombreProducto, marca, categoría, etc., para describir cada producto.
   - Se utiliza como dimensión en los contextos de compra y venta.

Uso de Llaves Surrogadas

- Se han empleado llaves surrogadas (idCompra, idVenta, idProveedor, idSucursal, idCliente, idVendedor, idProducto) en lugar de claves naturales para mejorar la eficiencia y la integridad referencial del modelo.
- Estas llaves proporcionan identificadores únicos para cada registro en las tablas, facilitando así las operaciones de consulta y la gestión de relaciones entre las entidades.

CONCLUSIÓN

Mediante el uso de SSIS, se ha logrado implementar un proceso eficiente y escalable para la integración de datos. Este manual técnico proporciona una visión completa de cada etapa del proceso ETL, desde la extracción inicial hasta la carga final en las tablas de destino. Al seguir este enfoque estructurado, se garantiza la consistencia y calidad de los datos, lo que facilita su posterior análisis y utilización en diversas aplicaciones empresariales.