**Fases de ETL:**

Primero, hicimos el DataWareHouse utilizando el modelo estrella. En este modelo tendremos una tabla hechos, la cual denominamos “Fact\_Payment” , la cual contiene las métricas o datos cuantitativos del negocio y distintas tablas dimensiones que contienen atributos descriptivos que se utilizan para filtrar y agrupar los datos en la tabla de hechos

Diagrama, Esquemático

Descripción generada automáticamente

**Creación de vistas para el ETL:**

Para facilitar el proceso **ETL** se crearon **vistas** que preparan los datos para su transformación. Por ejemplo, la tabla **Store** puede haber sido una de las dimensiones que se cargaron en el DataWarehouse. Estas vistas permiten organizar los datos de las bases de datos transaccionales (como Sakira en este caso) para que puedan ser transformados de manera eficiente y evitar inconsistencias durante la carga al DataWarehouse.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

**El origen de los datos:**

El primer paso del ETL es la **extracción** de los datos desde el sistema de origen. En la imagen correspondiente, se muestra cómo se seleccionan los datos desde la base de datos transaccional (Sakira, en este caso) u otros sistemas. Se eligen las tablas y columnas relevantes que se utilizarán en el DataWarehouse. Este paso asegura que solo se extraiga la información necesaria para el análisis.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Selección de datos para la transformación:**

Después de la extracción, se eligen los campos específicos de las tablas originales que serán **transformados** para adaptarse al modelo del DataWarehouse. Esto puede incluir la limpieza de datos, convertir formatos de fecha, normalización de texto, etc. En este caso, la imagen muestra cómo se seleccionan estos campos y se prepara la transformación.

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

**Uso de LookUp para evitar duplicación de SKeys:**

Durante la **transformación** en el proceso ETL, es importante que las claves de sustitución o **Surrogate Keys (SKey)** no se repitan. Para ello, se utiliza la operación **LookUp**, que verifica si una clave ya existe en el DataWarehouse. Si no existe, se genera una nueva clave; si existe, se reutiliza. Esto asegura la unicidad de los registros en las tablas dimensiones y hechos, evitando errores en la consolidación de datos.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Asignación al destino (DataWarehouse):**

En el paso de **carga**, los datos transformados se asignan a las tablas de destino en el DataWarehouse. En la imagen correspondiente, se ve cómo se eligen las columnas de la base de datos original (por ejemplo, la base de datos Sakira) y se mapean a las columnas correspondientes en el DataWarehouse. Cada columna de la base de datos de origen tiene su destino definido en el DataWarehouse, asegurando que la estructura de datos se mantenga consistente.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Prueba del proceso ETL:**

Una vez configurado el proceso ETL, se realiza una **prueba** para asegurarse de que el proceso funciona correctamente. Esto incluye verificar que los datos se hayan cargado en las tablas adecuadas, que los datos transformados sean precisos y que no haya errores o fallos durante el proceso. La prueba garantiza que el flujo de trabajo está completo y listo para ser automatizado en futuros ciclos de ETL.Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Desarrollo del cubo en Visual Studio:**

Después de la carga de datos en el DataWarehouse, se procedió a desarrollar un **cubo de datos** en Visual Studio. Un cubo de datos es una estructura que permite realizar análisis multidimensionales de los datos cargados en el DataWarehouse. El cubo organiza los datos en dimensiones y medidas, lo que facilita la creación de informes y análisis ad hoc. La estructura del cubo incluye las dimensiones definidas en el modelo estrella (como Tienda, Fecha, Cliente) y las métricas de la tabla de hechos (como el total de pagos o ventas).

Una captura de pantalla de una computadora

Descripción generada automáticamente con confianza media

Una vez que el cubo está construido, los datos pueden ser consultados directamente en **SQL Server**. En la imagen correspondiente, se muestra un ejemplo de cómo las consultas pueden extraer información del cubo para su análisis. Los usuarios pueden escribir consultas MDX (Multidimensional Expressions) o SQL para analizar los datos desde diferentes perspectivas, como las ventas por región o el rendimiento de productos específicos. Este paso permite realizar análisis de gran volumen de datos de manera rápida y eficiente.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Integración con Power BI:**

Finalmente, los datos del cubo se integran con herramientas de visualización como **Power BI**. En Power BI, los datos se pueden presentar en gráficos interactivos, dashboards, y reportes dinámicos. La integración con Power BI permite que los usuarios finales puedan interactuar con los datos de manera sencilla, filtrando y explorando la información según sus necesidades. Este paso cierra el ciclo de análisis al convertir los datos procesados en información accesible para la toma de decisiones.

