

UNIVERSIDAD GALILEO

Postgrado en Análisis y Predicción de Datos

Curso: Diseño y Construcción de Data Warehouses Horario: Lun: 18:00 – 21:00

Tutor: Msc. José Rolando Lucero Morataya



Otto Francisco Chamo Cheley – 19001395
Edgar Geovany Ocaña Orozco – 24010004
Jose Eduardo Calderon Lopez - 14001764
Andrea María Hernández Marroquín – 240110074

Guatemala, 05 de abril de 2024

ÍNDICE

Contenido

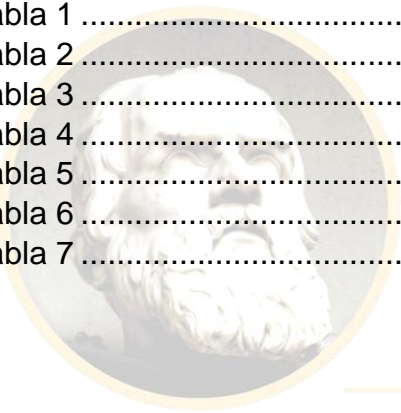
| | |
|--|----|
| INDICE DE IMAGENES | 3 |
| INDICE DE TABLAS | 4 |
| INTRODUCCIÓN: | 5 |
| RESUMEN EJECUTIVO | 8 |
| OBJETIVO: | 8 |
| METODOLOGÍA: | 8 |
| CONCLUSIONES: | 8 |
| RECOMENDACIONES: | 9 |
| MODELO ENTIDAD RELACIÓN | 14 |
| LIMPIEZA DE LOS DATOS EN TABLEU PREP: | 15 |
| ANÁLISIS EXPLORATORIO: | 22 |
| RENDIMIENTO GENERAL DE VENTAS: | 23 |
| RENDIMIENTO POR REGIÓN: | 23 |
| RENDIMIENTO POR CATEGORÍA DE PRODUCTO: | 24 |
| ANÁLISIS DE CLIENTES: | 25 |
| ANÁLISIS DE MODOS DE ENVÍO: | 25 |
| CONCLUSIONES | 26 |
| RECOMENDACIONES | 27 |
| REFERENCIAS BIBLIOGRAFICAS | 28 |

INDICE DE IMAGENES

| | |
|----------------------|----|
| Ilustración 1 | 14 |
| Ilustración 2 | 15 |
| Ilustración 3 | 16 |
| Ilustración 4 | 16 |
| Ilustración 5 | 17 |
| Ilustración 6 | 18 |
| Ilustración 7 | 18 |
| Ilustración 8 | 19 |
| Ilustración 9 | 19 |
| Ilustración 10 | 20 |
| Ilustración 11 | 22 |
| Ilustración 12 | 23 |
| Ilustración 13 | 24 |
| Ilustración 14 | 24 |
| Ilustración 15 | 25 |
| Ilustración 16 | 25 |

INDICE DE TABLAS

| | |
|---------------|----|
| Tabla 1 | 9 |
| Tabla 2 | 10 |
| Tabla 3 | 11 |
| Tabla 4 | 11 |
| Tabla 5 | 12 |
| Tabla 6 | 12 |
| Tabla 7 | 20 |



INTRODUCCIÓN:

El almacenamiento, gestión y análisis eficaz de datos son aspectos fundamentales en el ámbito empresarial actual. En un entorno donde la información es un activo valioso, la capacidad de aprovecharla de manera efectiva se convierte en un factor crítico para el éxito organizacional. En este contexto, el diseño e implementación de un Data Warehouse emerge como una solución clave para consolidar y estructurar datos de diversas fuentes en un repositorio centralizado, facilitando su acceso y análisis.

El presente proyecto se centra en el diseño de un Data Warehouse a partir de un set de datos, con el objetivo de proporcionar una plataforma robusta para el análisis y la toma de decisiones informadas. Este proceso implica la creación de tablas dimensionales y de referencias que representen de manera precisa las entidades y relaciones clave presentes en los datos originales. Para garantizar la calidad y coherencia de la información almacenada, se lleva a cabo un exhaustivo proceso de limpieza y preparación de datos, utilizando herramientas como Tableau Prep.

Además de la preparación de los datos para su inserción en el Data Warehouse, se realiza una etapa adicional de presentación en Tableau, lo que permite visualizar y analizar los datos de manera intuitiva y dinámica. Esta combinación de diseño de estructuras de datos y presentación visual facilita la comprensión y el aprovechamiento de la información almacenada, contribuyendo así a la generación de conocimientos y la identificación de insights relevantes para la organización.

En resumen, este proyecto aborda la creciente necesidad de gestionar y utilizar datos de manera efectiva, proporcionando una solución integral que combina diseño de Data Warehouse, limpieza de datos y visualización interactiva. A través de este enfoque, se espera contribuir significativamente a la capacidad de la organización para aprovechar su información de manera estratégica y competitiva.

MARCO TEÓRICO:

En el contexto actual de la era digital y la economía basada en datos, las organizaciones enfrentan el desafío de gestionar grandes volúmenes de información de manera efectiva para impulsar la toma de decisiones informadas y estratégicas. En este sentido, el concepto de Data Warehouse ha surgido como una solución fundamental para la integración, almacenamiento y análisis de datos empresariales.

El Data Warehouse se define como un sistema de gestión y almacenamiento de datos que nos permite almacenar, administrar y transformar grandes volúmenes de información de diferentes fuentes para ayudarnos en la toma de decisiones.

Este tipo de herramientas nos ayudan a aumentar la productividad y optimizar los recursos de nuestra empresa ayudando a la realización de informes que ayudarán a la empresa a reducir sus tiempos de análisis de datos.

A mantener registros históricos ya que en ellos se pueden almacenar grandes cantidades de información y así realizar predicciones futuras y crear comparativas, en base a las tendencias registradas.

A tomar mejores decisiones y en menor cantidad de tiempo. entre otras.

El concepto de Data Warehouse se originó en la década de 1980, cuando William H. Inmon y Ralph Kimball desarrollaron enfoques distintos pero complementarios para su diseño y construcción. El enfoque de Inmon, conocido como el modelo de la pirámide, se centra en la integración de datos de múltiples fuentes en un almacén centralizado antes de la presentación a los usuarios finales. Por otro lado, el enfoque de Kimball, conocido como la metodología de la arquitectura dimensional, se centra en la creación de estructuras de datos dimensionales optimizadas para consultas analíticas.

Además de los enfoques de diseño, el éxito de un Data Warehouse también depende de la calidad y limpieza de los datos. La limpieza de datos implica la detección y corrección de errores, la estandarización de formatos y la eliminación de duplicados, garantizando así la integridad y consistencia de la información almacenada.

En el contexto actual, el crecimiento exponencial de datos no estructurados, la proliferación de fuentes de datos y el surgimiento de nuevas tecnologías como el procesamiento en la nube y el análisis predictivo, plantean desafíos y oportunidades adicionales para el diseño y gestión de Data Warehouses..



RESUMEN EJECUTIVO

Proyecto: Diseño de Data Warehouse

OBJETIVO:

El presente proyecto se ha llevado a cabo con el objetivo de diseñar un Data Warehouse utilizando un set de datos proporcionado como punto de partida. Se buscó estructurar tablas dimensionales y de referencias de datos basadas en los archivos previamente entregados.

METODOLOGÍA:

Para lograr los objetivos establecidos, se implementó una metodología que incluyó las siguientes etapas:

1. Análisis de los datos proporcionados para identificar las dimensiones y relaciones clave.
2. Diseño y creación de tablas dimensionales y de referencias de datos.
3. Limpieza y preparación de los datos utilizando herramientas como Tableau Prep para garantizar la calidad y coherencia de estos.
4. Integración de los datos limpios en las dimensiones diseñadas.
5. Presentación de los datos en Tableau para análisis y visualización adicionales.
6. Limpieza adicional de datos para facilitar la inserción en la tabla de referencias.

Resultados Clave:

Los resultados obtenidos muestran la creación exitosa de un Data Warehouse con tablas dimensionales y de referencias que reflejan de manera precisa los datos originales. La presentación de datos en Tableau ha proporcionado una visualización efectiva de los datos limpios, facilitando el análisis y la comprensión.

CONCLUSIONES:

En conclusión, este proyecto ha demostrado la viabilidad y eficacia del diseño de un Data Warehouse a partir de un set de datos dado. La combinación de limpieza de datos

y presentación en Tableau ha mejorado significativamente la utilidad y la accesibilidad de la información almacenada.

RECOMENDACIONES:

Se recomienda continuar explorando las capacidades de Tableau y otras herramientas de visualización para maximizar el valor del Data Warehouse. Además, se sugiere establecer procesos sólidos de mantenimiento y actualización de datos para garantizar la relevancia y precisión continua del Data Warehouse.

Este resumen ejecutivo proporciona una visión general de los principales aspectos abordados en el proyecto. Para obtener detalles adicionales sobre la metodología, resultados y conclusiones, se recomienda revisar el documento completo.

DISEÑO DEL DATA WAREHOUSE

Para la realización de este proyecto se creó una base de datos llamada **STORE** que está compuesta por las siguientes tablas:

1. **Tabla Dim_Date:** En esta tabla se decidieron hacer cambios en la declaración del tipo de dato en los campos quedando así:

Tabla 1

| Tabla Original | Tabla Modificada |
|---|--|
| Date_Dimension | Dim_Date |
| date_key smallint not null, full_date smalldatetime, day_of_week tinyint, day_num_in_month tinyint, day_num_overall smallint, day_name varchar(9), day_abbrev char(3), weekday_flag char(1), week_num_in_year tinyint, week_num_overall smallint, | date_key INT PRIMARY KEY, full_date DATE, day_of_week TINYINT, day_num_in_month TINYINT, day_num_overall INT, day_name VARCHAR(9), day_abbrev CHAR(3), weekday_flag CHAR(7), week_num_in_year TINYINT, week_num_overall INT, |

| | |
|--|---|
| week_begin_date smalldatetime, week_begin_date_key smallint, month tinyint, month_num_overall smallint, month_name varchar(9), month_abbrev char(3), quarter tinyint, year smallint, yearmo int, fiscal_month tinyint, fiscal_quarter tinyint, fiscal_year smallint, last_day_in_month_flag char(1), same_day_year_ago_date smalldatetime, primary key (date_key) ; | week_begin_date DATE, week_begin_date_key INT, month TINYINT, month_num_overall INT, month_name VARCHAR(9), month_abbrev VARCHAR(3), quarter TINYINT, year INT, yearmo INT, fiscal_month TINYINT, fiscal_quarter TINYINT, fiscal_year INT, last_day_in_month_flag ENUM('Not Month End', 'Month End'), same_day_year_ago_date DATE); |
|--|---|

2. Tabla Dim_Customer:

Se incluyó el campo **segmento** en la tabla **dim_customers** dado que se consideró que forma parte de la estructura de atributos de los clientes. Este atributo se añadió con el fin de proporcionar una mayor granularidad en la segmentación de los clientes, permitiendo así un análisis más detallado y una mejor comprensión de las distintas categorías y características que los definen. A continuación, se presenta la tabla final con todos sus campos:

Tabla 2

| |
|---------------------|
| Tabla Final |
| Dim_Customer |

| | |
|-----------------------------|---------------|
| SK_Customer | INT |
| AUTO_INCREMENT PRIMARY KEY, | |
| Customer_ID | VARCHAR(8), |
| Customer_Name | VARCHAR(255), |
| Segment | VARCHAR(50), |
| Timestamp | DATETIME |

3. Tabla Dim_Product:

Tabla 3

| Tabla Final | |
|--------------|--------------------|
| Dim_Product | |
| SK_Product | INT AUTO_INCREMENT |
| PRIMARY KEY, | |
| Product_ID | VARCHAR(15), |
| Product_Name | VARCHAR(255), |
| Category | VARCHAR(50), |
| Sub_Category | VARCHAR(50), |
| Timestamp | DATETIME |

4. **Tabla Dim_Ship_Mode:** Se creo la tabla Dim_Ship_Mode que contiene la información del modo de envío, a continuación, se muestra el diseño de la tabla final:

Tabla 4

| Tabla Final | |
|-----------------------------|--------------|
| Dim_Ship_Mode | |
| SK_Ship_Mode | INT |
| AUTO_INCREMENT PRIMARY KEY, | |
| Ship_Mode | VARCHAR(50), |

| |
|--------------------|
| Timestamp DATETIME |
|--------------------|

5. **Tabla_Dim_Location:** Se cambió el nombre de se colocó location en vez de la palabra geography quedando la tabla de la siguiente manera:

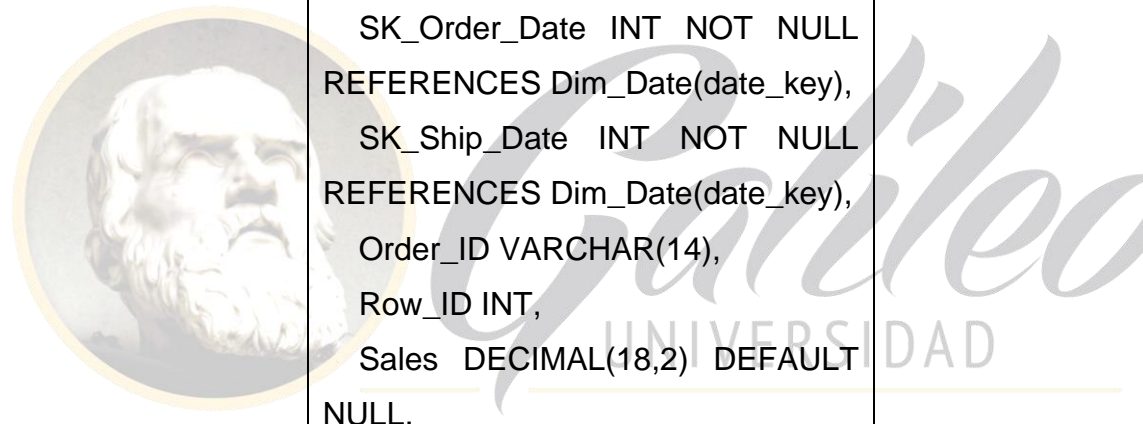
Tabla 5

| Tabla Final | |
|-----------------------------|-----|
| sales_staging | |
| SK_Location | INT |
| AUTO_INCREMENT PRIMARY KEY, | |
| Country VARCHAR(50), | |
| Region VARCHAR(10), | |
| State VARCHAR(50), | |
| City VARCHAR(255), | |
| Postal_Code INT, | |
| Timestamp DATETIME | |

6. **Tabla de Hechos Sales_Fact:** Se decidió hacer una tabla de referencias de un nivel ya que al utilizar una no es recomendable crear una tabla de dimensiones de tipo snowflake en una tabla de hechos principalmente por razones de rendimiento y simplicidad en el diseño de la base de datos, así como también se procedió a crear las referencias SK_Order_Date y SK_Ship_Date como se requirió en este ejercicio, a modo que posteriormente se pudieran guardar el campo date_key dos veces, por lo que el diseño quedo de la siguiente forma:

Tabla 6

| Tabla Final |
|-------------|
| Sales_Fact |



```

SK_Customer INT NOT NULL
REFERENCES
Dim_Customer(SK_Customer),
    SK_Product INT NOT NULL
REFERENCES
Dim_Product(SK_Product),
    SK_Ship_Mode INT NOT NULL
REFERENCES
Dim_Ship_Mode(SK_Ship_Mode),
    SK_Location INT NOT NULL
REFERENCES
Dim_Location(SK_Location),
    SK_Order_Date INT NOT NULL
REFERENCES Dim_Date(date_key),
    SK_Ship_Date INT NOT NULL
REFERENCES Dim_Date(date_key),
    Order_ID VARCHAR(14),
    Row_ID INT,
    Sales DECIMAL(18,2) DEFAULT
NULL,
    Quantity INT DEFAULT NULL,
    Discount DECIMAL(18,2)
DEFAULT NULL,
    Profit DECIMAL(18,2) DEFAULT
NULL

```

A todas las tablas se les agregó la palabra Dim para identificar las dimensiones con excepción de la tabla de hechos; y a las llaves primarias se les colocó la palabra SK_ para poderlas identificar.

MODELO ENTIDAD RELACIÓN

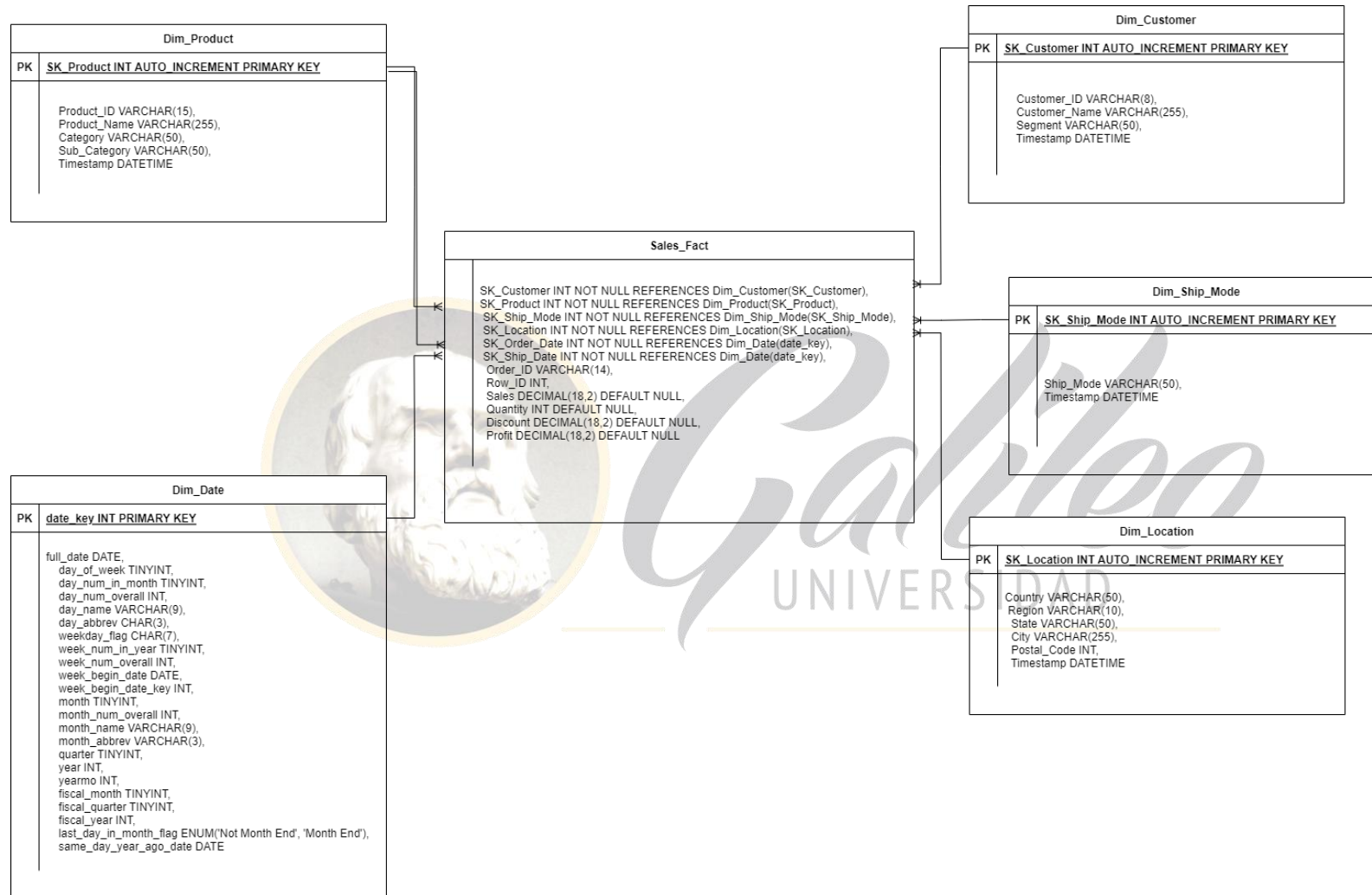


Ilustración 1

LIMPIEZA DE LOS DATOS EN TABLEU PREP:

Previo a la realización del Data Warehouse consideramos necesaria la limpieza de los datos proporcionados para así asegurar la calidad y la consistencia de los datos, los cuales permitirán un mejor rendimiento a la hora de procesar consultas y generar informes.

Los cambios que se realizarán a la información fueron los siguientes:

1. Se agregó guión bajo (_) a cada campo del archivo que tuviera más de dos palabras del archivo **SuperStoreOutput.csv**

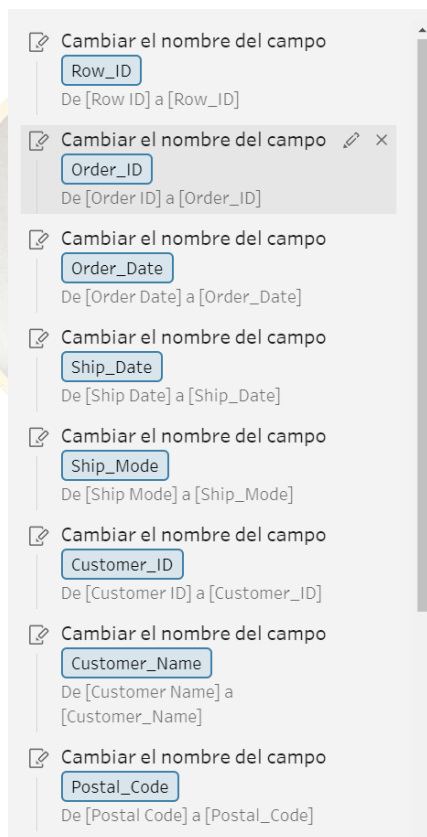


Ilustración 2

2. Posteriormente del archivo dado se seleccionaron los atributos de cada una de las tablas que previamente diseñamos en nuestra Base de Datos **STORE**.

| | | | | | |
|---|-------|-------------------------------|--------------------|------------------|-------------|
| Dim_Customer 3 campos 793 filas | | | Filtrar valores... | | |
| Configuración | | | Cambios (0) | | |
| Campos adicionales | | | Campos agrupados | | |
| Arrastre los campos para agregarlos o agruparlos. | | | | | |
| <input type="text" value="Buscar"/> | | | | | |
| <input type="button" value="Añadir todo"/> <input type="button" value="Eliminar todo"/> | | | | | |
| Abc | GROUP | Category | Abc | GROUP | Abc |
| Abc | GROUP | City | Customer_ID | Customer_Name | Segment |
| Abc | GROUP | Country | TS-21370 | Todd Sumrall | Corporate |
| # | SUM | Discount | JK-15370 | Jay Kimmel | Consumer |
| 🕒 | GROUP | Order_Date | ML-17410 | Maris LaWare | Consumer |
| Abc | GROUP | Order_ID | RK-19300 | Ralph Kennedy | Consumer |
| # | SUM | Postal_Code | SB-20185 | Sarah Brown | Consumer |
| Abc | GROUP | Product_ID | RB-19435 | Richard Bierner | Consumer |
| Abc | GROUP | Product_Name | EM-14200 | Evan Minnotte | Home Office |
| # | SUM | Profit | BW-11110 | Bart Watters | Corporate |
| # | SUM | Quantity | DG-13300 | Deirdre Greer | Corporate |
| Abc | GROUP | Region | PT-19090 | Pete Takahito | Consumer |
| # | SUM | Row_ID | MC-17605 | Matt Connell | Corporate |
| # | SUM | Sales | HR-14830 | Harold Ryan | Corporate |
| 🕒 | GROUP | Ship_Date | CS-12490 | Cindy Schnelling | Corporate |
| Abc | GROUP | Ship_Mode | BC-11125 | Becky Castell | Home Office |
| Abc | GROUP | State | EB-14170 | Evan Bailliet | Consumer |
| Abc | GROUP | Sub_Category | EJ-13720 | Ed Jacobs | Consumer |
| # | SUM | Cantidad de filas (agregadas) | GB-14530 | George Bell | Corporate |
| | | | DM-13345 | Denise Monton | Corporate |
| | | | DM-13015 | Darrin Martin | Consumer |

Ilustración 3

- Luego de separar nuestros atributos en las diferentes tablas, se procedió a agregar el atributo **TIMESTAMP** de tipo DATETIME a las tablas **Dim_Location**, **Dim_Customer**, **Dim_Ship_Mode** y **Dim_product**.

| Cambios (1) | | |
|--|-----------------|---------|
| <input type="button" value="Campo calculado"/> | | |
| <input type="button" value="Timestamp"/> | | |
| <input type="button" value="NOW()"/> | | |
| Tipo | Nombre de campo | Cambios |
| Abc | Customer_ID | |
| Abc | Customer_Name | |
| Abc | Segment | |
| 🕒 | Timestamp | 🕒 |

Ilustración 4

- Para la realización de la Tabla **Dim_Date** para poder limpiar la información se crearon las columnas llamadas **date_key_new** y **week_begin_date_new** las cuales contenían los datos almacenados de las columnas **date_key** y **week_begin_date**

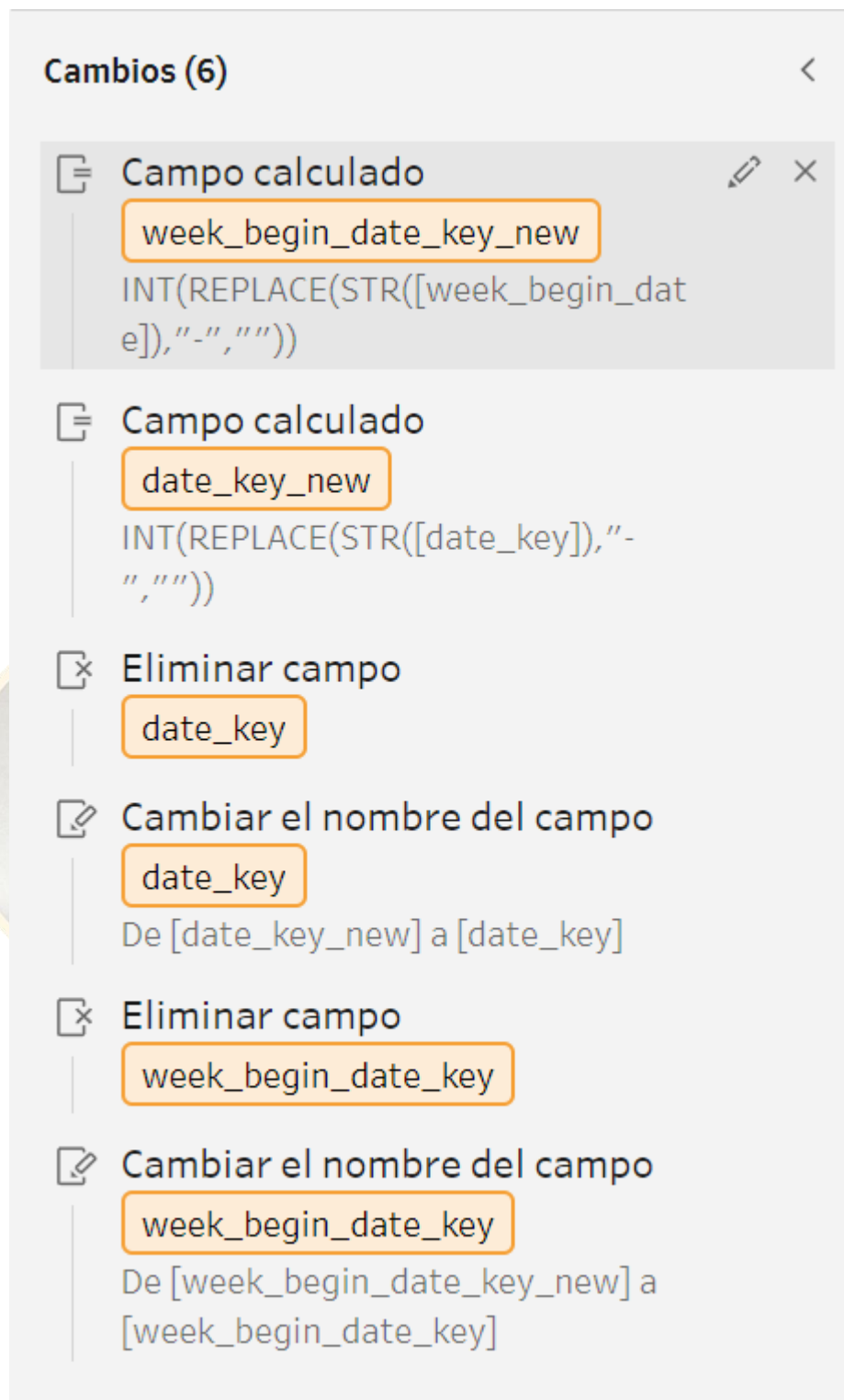


Ilustración 5

5. Posterior a la creación de dichas columnas se procedió a reemplazar los guiones que aparecían y a reemplazarlos con por espacios vacíos y luego convertir ese dato en un entero.

Editar campo

Nombre de campo

date_key_new

`INT (REPLACE (STR ([date_key]) , "-", ""))`

Ilustración 6

6. Como siguiente paso se procedió a eliminar los campos **date_key** y **week_begin_date** y se reemplazaron por las nuevas columnas **date_key_new** y **week_begin_date_new**

Eliminar campo

date_key

Cambiar el nombre del campo

date_key

De [date_key_new] a [date_key]

Ilustración 7

7. Luego se procedió a cambiarle los nombres a los campos **date_key_new** y **week_begin_date_new** por **date_key** y **week_begin_date**.

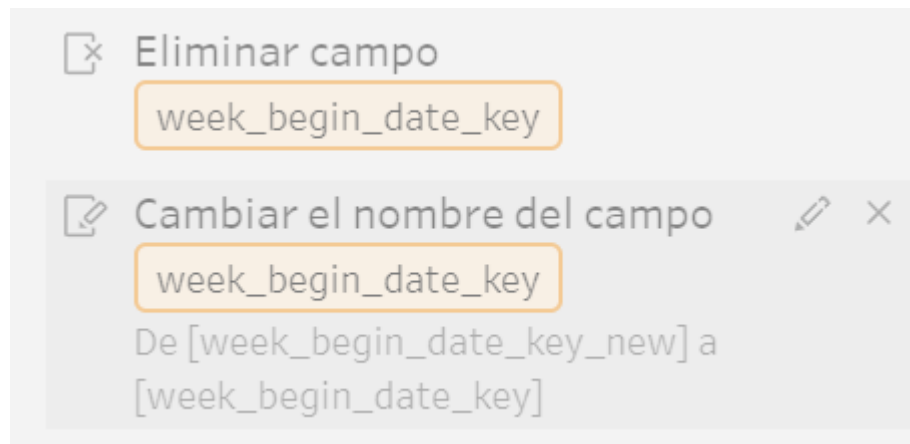


Ilustración 8

8. Posterior a la limpieza de los datos se procedió a realizar los outputs en los que se realiza la conexión a la Base de Datos **STORE**

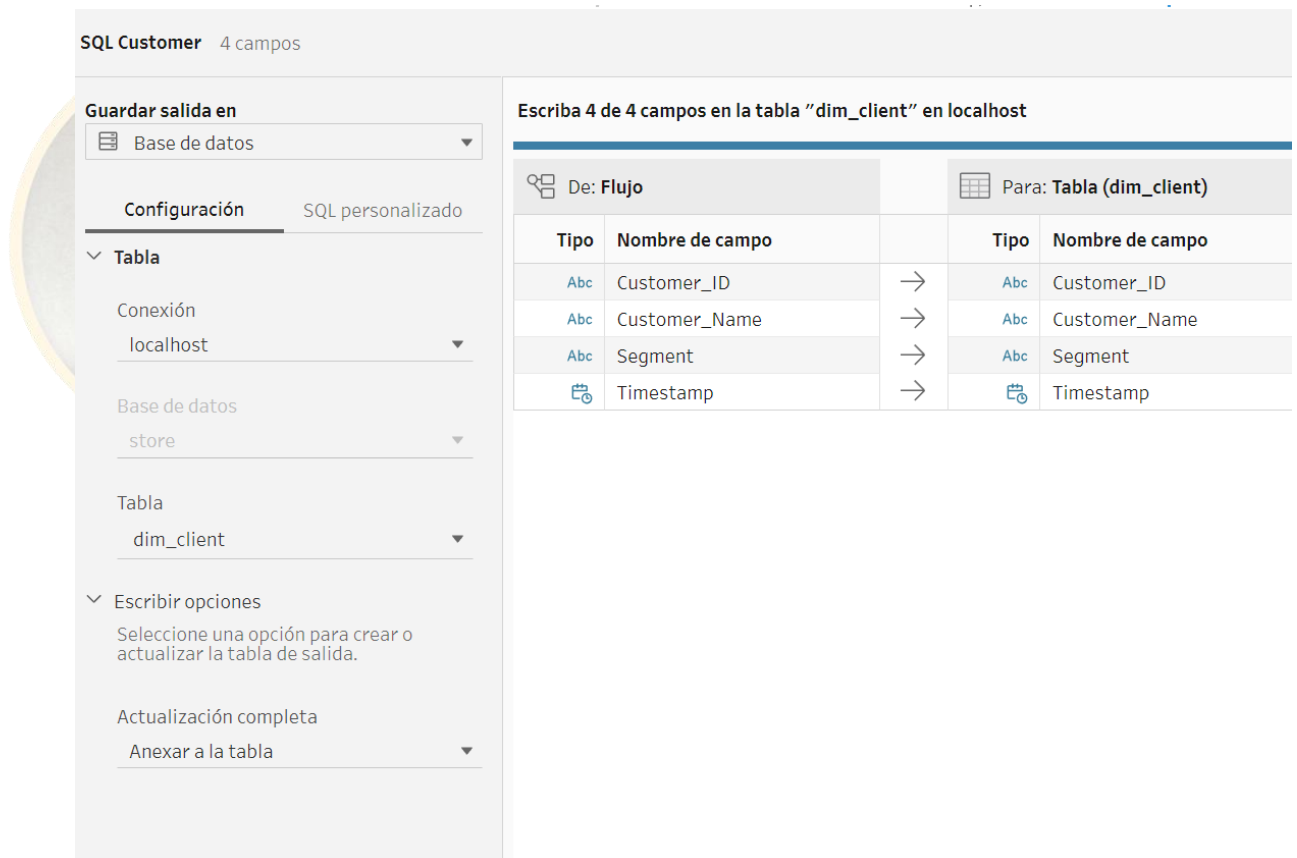


Ilustración 9

9. Quedando como resultado final el siguiente esquema en Tableau

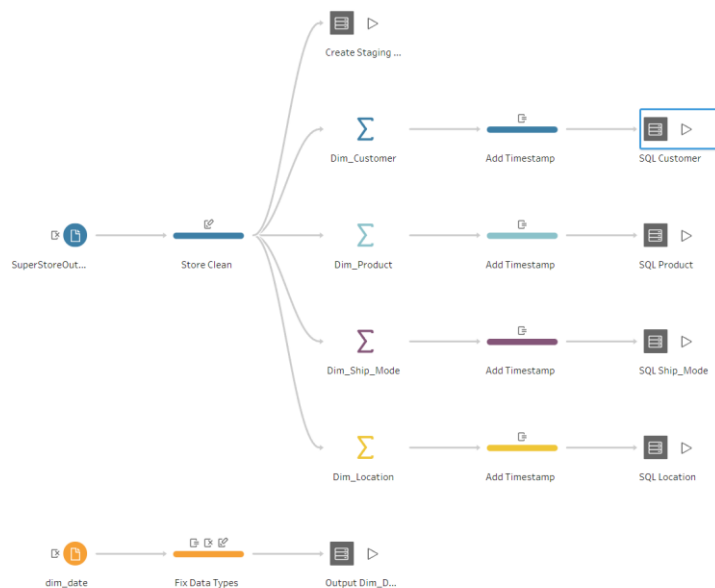


Ilustración 10

10. Posterior a la creación de los OUTPUTS de las tablas de nuestra base de datos se procedió a crear un OUTPUT de la tabla Tabla de Staging (Creada por Tableau) sales_staging

Tabla 7

| Tabla |
|--|
| sales_staging |
| <pre> `Row_ID` bigint DEFAULT NULL, `Order_ID` text, `Order_Date` datetime DEFAULT NULL, `Ship_Date` datetime DEFAULT NULL, `Ship_Mode` text, `Customer_ID` text, `Customer_Name` text, `Segment` text, `Country` text, </pre> |

```
`City` text,  
`State` text,  
`Postal_Code` bigint DEFAULT  
NULL,  
`Region` text,  
`Product_ID` text,  
`Product_Name` text,  
`Category` text,  
`Sub_Category` text,  
`Sales` double DEFAULT NULL,  
`Quantity` bigint DEFAULT NULL,  
`Discount` double DEFAULT NULL,  
`Profit` double DEFAULT NULL
```

11. Utilizando la información almacenada en la tabla sales_staging se procedió a realizar join con cada una de las dimensiones, sin embargo, previo a la realización de este paso se procedió a realizar una limpieza a los datos y realizar cambios como el cambio a tipo DATE en las columnas order_date y ship_date.
12. Se procedió a la eliminación de los campos **customer_id**, **customer_id_1**, **product_id**, **full_date**, **Order_Date**, **date_of_week**, **ship_date** y se dejó el Campo SK_Customer, SK_Product, y se renombró los campos date_key y order_date por SK_Order_Date y SK_ship_date a modo de dejar nuestros campos como la tabla de hechos que diseñamos y utilizar dos campos de dimensiones de tiempo

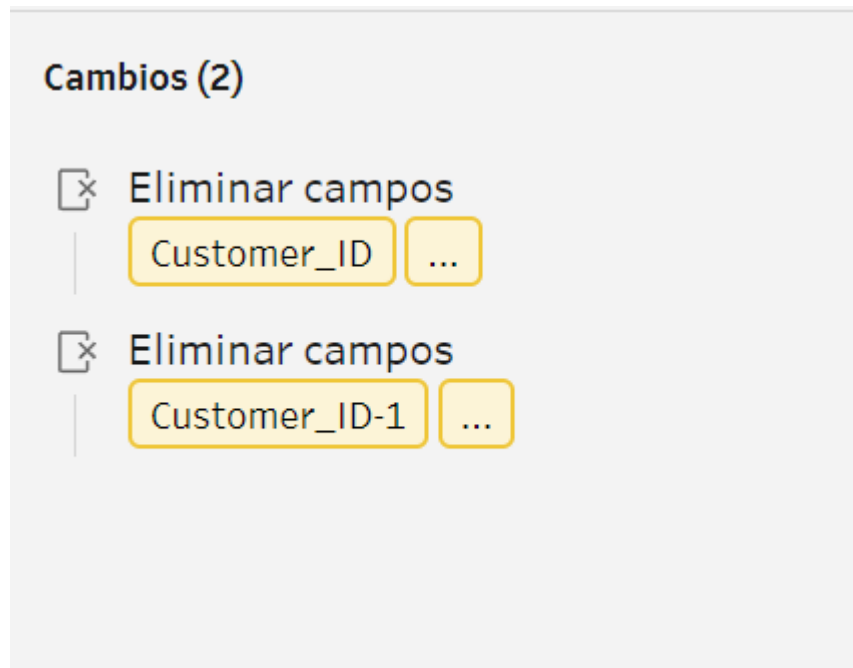


Ilustración 11

13. Posterior al tratamiento de la información se procedió a la realización de un OUTPUT en el que se realiza una conexión con la base de Datos STORE para realizar los INSERTS de los datos a nuestra tabla de hechos.

14.

¿Cómo utilizaremos el campo Order_ID?

En cuanto a la utilización del Campo Order_ID se decidió colocarla en la tabla de hechos

ANÁLISIS EXPLORATORIO:

Este análisis exploratorio se basa en la información proporcionada con la información del Datawarehouse trabajado en el proyecto, apoyado por la tabla de hechos y sus respectivas tablas dimensionales. Se analizan las tendencias generales de ventas, el rendimiento por región, categoría de producto, cliente y modo de envío.

RENDIMIENTO GENERAL DE VENTAS:

Se observa una estacionalidad clara, con mayores ventas en septiembre, noviembre y diciembre, siendo noviembre el mes con mayor volumen.

Febrero es el mes con menor cantidad de ventas, seguido por enero y abril.

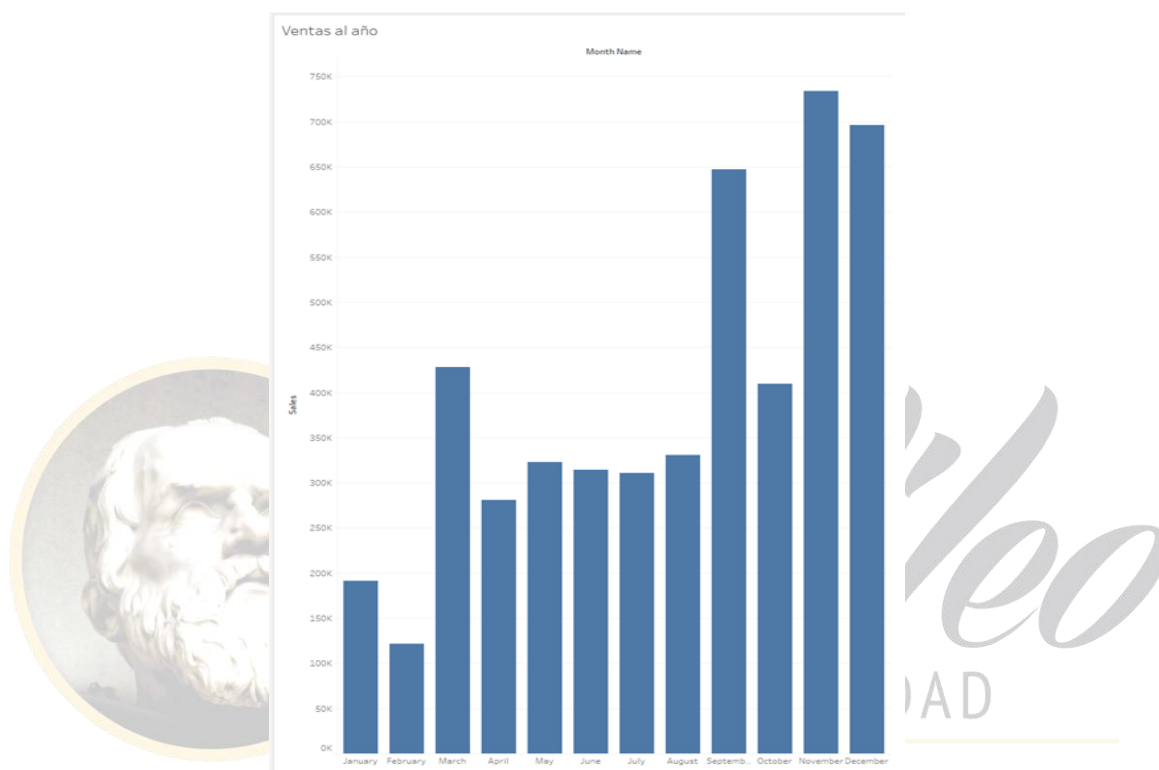


Ilustración 12

RENDIMIENTO POR REGIÓN:

Las costas de Estados Unidos concentran la mayor actividad, con California, Virginia, Texas, Florida, Georgia, Carolina del Norte, Carolina del Sur y Nueva Jersey como los estados con mayor movimiento.

Las mayores compras se realizan en Nueva Jersey, Texas y California.

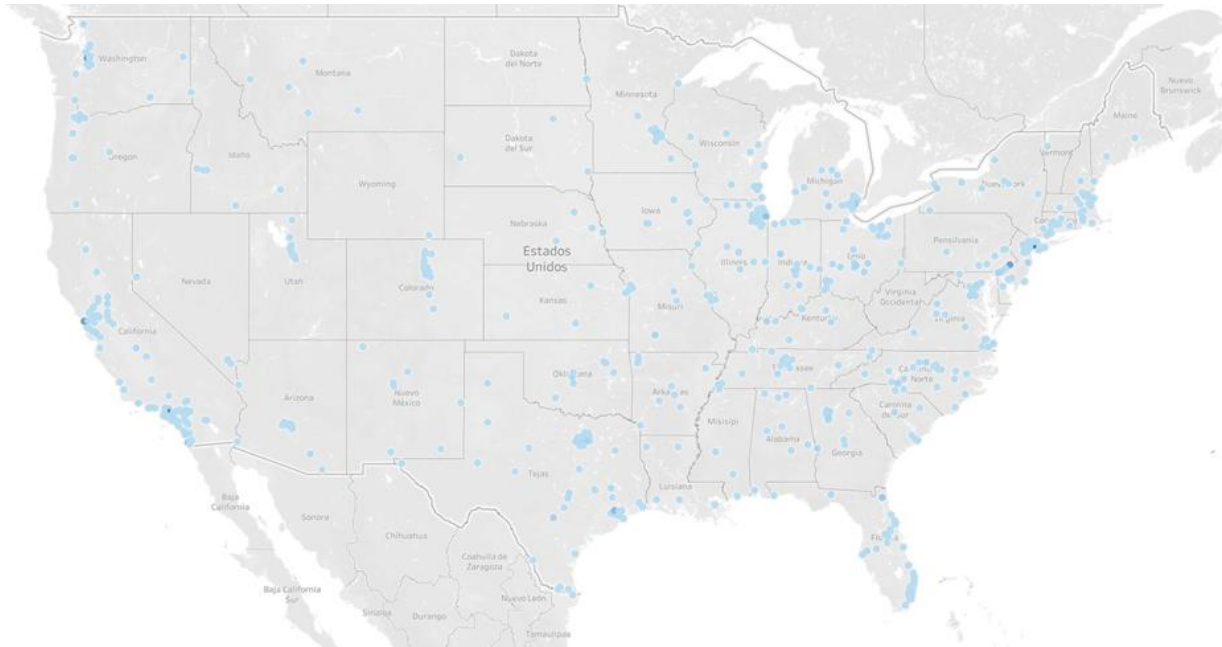


Ilustración 13

RENDIMIENTO POR CATEGORÍA DE PRODUCTO:

Tecnología es la categoría con mayor volumen de ventas, seguida por Furniture y Office Supplies.



Ilustración 14

ANÁLISIS DE CLIENTES:

Sean Miller es el cliente de mayor valor, seguido por Tamara Chand, Greg Tran y Raymond Buch.

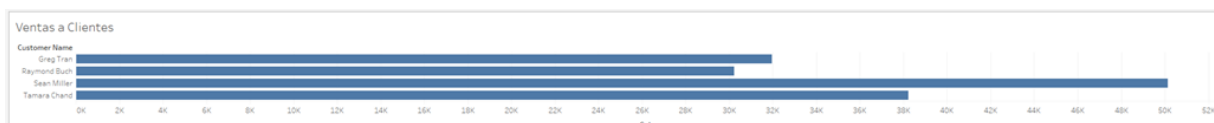


Ilustración 15

Las compras totales de los clientes de alto valor oscilan entre 29.879 y 50.148.

ANÁLISIS DE MODOS DE ENVÍO:

Standard Class es el modo de envío más utilizado, con un repunte en los meses de mayor venta.

Second Class es el segundo modo más popular, seguido por First Class y Same Day.

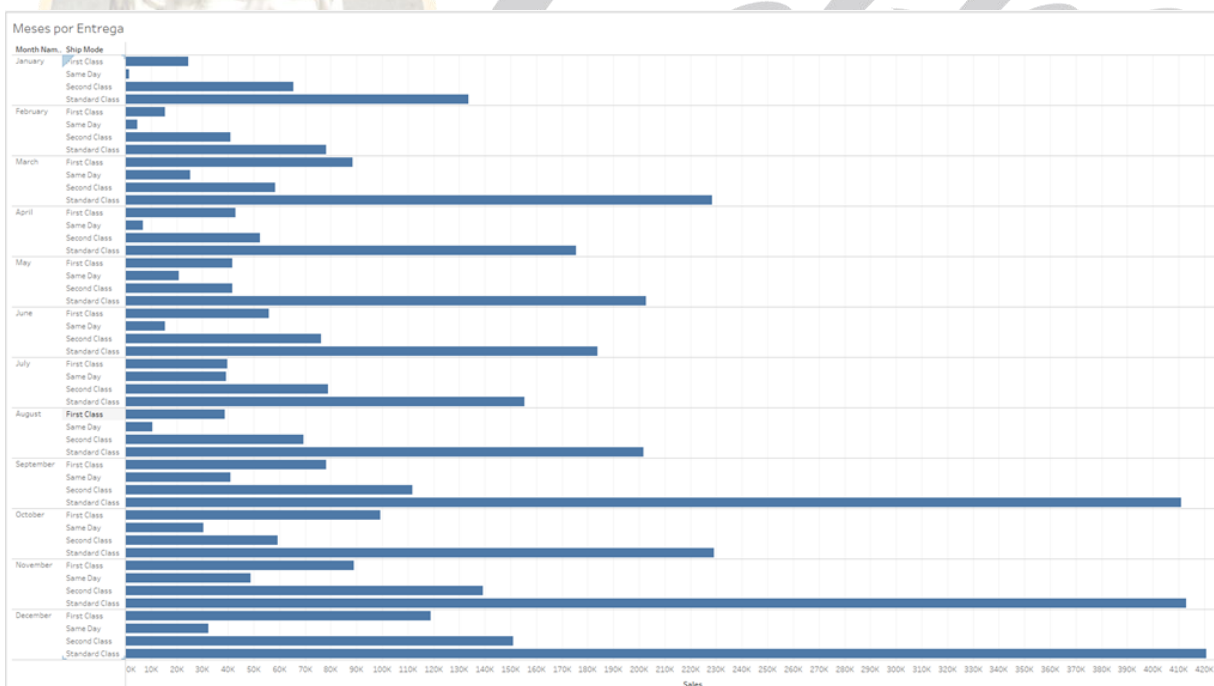


Ilustración 16

CONCLUSIONES

El proceso de diseño e implementación de un Data Warehouse ha proporcionado una serie de aprendizajes y resultados significativos. A través de este proyecto, se han identificado varios puntos clave que resumen la experiencia y las contribuciones de este esfuerzo:

1. Estructuración y modelado de datos: La creación de tablas dimensionales y de referencias ha sido fundamental para organizar y representar adecuadamente la información contenida en el set de datos original. Este proceso ha permitido definir relaciones claras entre las diferentes entidades, facilitando así el análisis y la extracción de conocimientos.

2. Limpieza y preparación de datos: La etapa de limpieza y preparación de datos ha demostrado ser una fase crítica pero necesaria en el proceso de diseño del Data Warehouse. La utilización de herramientas como Tableau Prep ha facilitado la identificación y corrección de inconsistencias y errores en los datos, asegurando la integridad y calidad de la información almacenada.

3. Presentación visual y análisis de datos: La visualización de datos en Tableau ha añadido un valor significativo al proyecto al proporcionar una interfaz interactiva y dinámica para explorar y comprender los datos. Esta capacidad de presentar información de manera intuitiva ha facilitado la identificación de patrones, tendencias y relaciones ocultas en los datos, proporcionando así datos valiosos para la toma de decisiones.

En conclusión, la realización de estos procesos nos ayudará a tomar decisiones, y mejorar el rendimiento de nuestros data Warehouse.

RECOMENDACIONES

1. Se recomienda que en la fase de diseño inicial de un Data Warehouse se dedique el tiempo y los recursos necesarios para garantizar que las relaciones entre las diferentes entidades estén claramente definidas y organizadas de manera efectiva.
2. La etapa de limpieza y preparación de datos es crucial para garantizar la calidad y la integridad de la información almacenada. Se recomienda continuar explorando herramientas y técnicas que puedan mejorar y automatizar este proceso, lo que puede incluir la implementación de soluciones de limpieza de datos más avanzadas o la integración de reglas y validaciones adicionales para detectar y corregir posibles errores de manera más eficiente.
3. La visualización de datos en Tableau ha demostrado ser una herramienta valiosa para el análisis y la presentación de información. Se sugiere ampliar el uso de esta plataforma para ofrecer capacidades de visualización más avanzadas y personalizadas, lo que podría incluir la creación de paneles interactivos y tableros personalizados para diferentes usuarios y departamentos.
4. Es importante realizar una evaluación periódica del rendimiento y la eficacia de los procesos de diseño e implementación de Data Warehouse.

REFERENCIAS BIBLIOGRAFICAS

Referencias

Amazon Web Services. (s/f de s/f de s/f). aws. Obtenido de ¿Cuál es la diferencia entre los procesos de ETL y ELT?: ¿Cuál es la diferencia entre los procesos de ETL y ELT?

Amazon Web Services. (s/f de s/f de s/f). aws. Obtenido de ¿Qué es el procesamiento analítico en línea (OLAP)?: [https://aws.amazon.com/es/what-is/olap/#:~:text=El%20procesamiento%20anal%C3%ADtico%20en%20l%C3%ADnea%20\(OLAP\)%20es%20una%20tecnolog%C3%ADa%20de,medidores%20inteligentes%20y%20sistemas%20internos.](https://aws.amazon.com/es/what-is/olap/#:~:text=El%20procesamiento%20anal%C3%ADtico%20en%20l%C3%ADnea%20(OLAP)%20es%20una%20tecnolog%C3%ADa%20de,medidores%20inteligentes%20y%20sistemas%20internos.)

Amazon Web Services. (s/f de s/f de s/f). aws. Obtenido de ¿Cuál es la diferencia entre OLAP y OLTP?: <https://aws.amazon.com/es/compare/the-difference-between-olap-and-oltp/#:~:text=Un%20sistema%20de%20OLAP%20le,procesar%20transacciones%20en%20tiempo%20real.>

Amazon Web Services. (s/f de s/f de s/f). AWS. Obtenido de ¿Qué es extracción, transformación y carga (ETL)?: <https://aws.amazon.com/es/what-is/etl/#:~:text=es%20AWS%20Glue%3F-,%C2%BFQu%C3%A9%20es%20ETL%3F,central%20llamado%20almacenamiento%20de%20datos.>

Castelán García , L., & Ocharán Hernández, J. (s/f de s/f de s/f). *Diseño de un Almacén de datos basado en Data Warehouse Engineerin Process (DWEPE) y HEFESTO*. Obtenido de Universidad Veracruzana: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/<https://www.uv.mx/mis/files/2012/11/Diseno-de-un-Almacen-de-datos.pdf>

Kimball, R. (17 de 09 de 2002). *Kimball Group*. Obtenido de Two Powerful Ideas: <https://www.kimballgroup.com/2002/09/two-powerful-ideas/>

Ross, M. (01 de 01 de 2004). *Kimball Group*. Obtenido de <https://www.kimballgroup.com/2004/01/data-warehouse-dining-experience/>