**Universidad Técnica Nacional**

**Sede Ciudad Quesada, San Carlos**

**Ingeniería del Software**

**ISW 912: Minería de Datos**

**Proyecto #1: ETL y Data Warehouse**

**Integrantes:**

**Hanzel Roberto Quirós Beteta**

**Jose Alí Rivas Gómez**

**Jose Arturo Quirós Araya**

**Docente:**

**Freddy Gerardo Rocha Boza**

**Periodo 2023**

**1er Cuatrimestre**

**Tabla de Contenidos**

**[INTRODUCCIÓN](#_Toc129513199)** [3](#_Toc129513199)

**[ENUNCIADO DEL PROBLEMA](#_Toc129513200)** [4](#_Toc129513200)

**[OBJETIVO GENERAL](#_Toc129513201)** [5](#_Toc129513201)

**[OBJETIVOS ESPECÍFICOS](#_Toc129513202)** [5](#_Toc129513202)

**[DESCRIPCIÓN DE LA SOLUCIÓN](#_Toc129513203)** [6](#_Toc129513203)

**[APARTADO DOCUMENTAL](#_Toc129513204)** [6](#_Toc129513204)

**[ESTRATEGIAS PARA LA CREACIÓN DE UN PROYECTO ETL](#_Toc129513205)** [6](#_Toc129513205)

**[ETAPAS DE ETL](#_Toc129513206)** [7](#_Toc129513206)

**[METODOLOGIAS](#_Toc129513207)** [8](#_Toc129513207)

**[APARTADO PRACTICO](#_Toc129513208)** [11](#_Toc129513208)

**[AMBIENTE DE PRODUCCIÓN](#_Toc129513209)** [11](#_Toc129513209)

**[AREA DE PREPARACIÓN (STAGING AREA)](#_Toc129513210)** [12](#_Toc129513210)

**[DATAWAREHOUSE](#_Toc129513211)** [16](#_Toc129513211)

**[CONCLUSIÓN](#_Toc129513212)** [26](#_Toc129513212)

**[BIBLIOGRAFÍA](#_Toc129513213)** [27](#_Toc129513213)

# **INTRODUCCIÓN**

Vivimos en un mundo altamente digitalizado donde se generan grandes cantidades de información, los datos son insumos que requieren ser procesados para su análisis, todo ésto se convierte en un reto.

La empresa Northwind posee una amplia Base de Datos, producto de sus operaciones diarias en venta de alimentos, sin embargo, el análisis de esta información es realizado mediante reportes estáticos y el uso de herramientas de ofimática como Excel, lo que limita el potencial de los resultados obtenidos por medio de su análisis.

Por medio del proceso ETL (Extract, Transform, Load) se procesarán los datos para la construcción de un Data Warehouse que sirva de herramienta para su estudio y construcción de tableros KPI’s, que representen de manera gráfica la información más importante para Northwind.

# **ENUNCIADO DEL PROBLEMA**

La construcción de un datawarehouse es un proceso que no debe ser tomado a la ligera debido a que su uso está orientado a la toma de decisiones, la incorrecta representación de la información podría ocasionar la ejecución de malas decisiones con gran impacto negativo a nivel operacional. Para este proceso, es necesario realizar un análisis profundo de la estructura en la Base de Datos de Producción, y luego, identificar aquella información que es relevante para las partes interesadas del negocio, encargadas de la dirección del mismo, con el fin de establecer el contenido esencial que se almacenará en el Data Warehouse. Para su construcción es necesario implementar un Staging Area, que sirva de insumo para acceder a toda la información relativamente sin alterar la base de datos en ambiente de producción.

Finalmente, se deberá realizar un proceso de ETL desde el Staging Area para cargar la información en el Data Warehouse y poder así analizar los datos.

# **OBJETIVO GENERAL**

Desarrollar un Data Warehouse usando la base de datos Northwind mediante el proceso ETL.

# **OBJETIVOS ESPECÍFICOS**

* Identificar las reglas de negocio para la construcción de un Data Warehouse.
* Investigar las mejores practicas acerca de un proceso de ETL y Data Warehouse.
* Evaluar la información durante la creación de un ambiente Staging Area (SA).
* Ejecutar el proceso ETL para la obtención de los datos.
* Transferir la información porterior al desarrollo del Data Warehouse.
* Representar mediante diagramas el modelo utilizado para el Data Warehouse.

# **DESCRIPCIÓN DE LA SOLUCIÓN**

## **APARTADO DOCUMENTAL**

### **ESTRATEGIAS PARA LA CREACIÓN DE UN PROYECTO ETL**

Los procesos ETL (Extract, Transform, Load) consisten en un conjunto de técnicas, herramientas y tecnologías que permiten extraer datos de varias fuentes, transformarlos de forma que sean veraces y útiles, y cargarlos en otros sistemas con el fin de que puedan ser accesibles por los niveles de la organización que así lo requieran con el fin de aportar a la toma de decisiones. A la hora de desarrollar un proceso de ETL el primer paso es el desarrollo de una estrategia en conjunto con el área de negocio de la empresa u organización para la cual se va a desarrollar el proceso. Para el desarrollo de una estrategia de ETL, el principal insumo que se va a requerir es el objetivo principal, por ejemplo: “Analizar las ventas y sus indicadores asociados para determinar los mejores vendedores y clientes”. Con ésto, el encargado o líder del proyecto deberá comenzar a realizar un análisis de las diferentes fuentes de información con las que cuenta la organización y determinar cuáles son sus características para dar comienzo al proyecto.

### **ETAPAS DE ETL**

#### **ANÁLISIS**

En la etapa de análisis el líder del proyecto (el cual deberá ser una persona con suficiente conocimiento técnico y con amplio conocimiento del área de negocio) en conjunto con las partes interesadas realizará la identificación y establecimiento de los objetivos de la organización. Con base en estos objetivos establecidos, generalmente por una alta dirección o personal de nivel gerencial, el líder del proyecto realizará una identificación de las posibles fuentes de datos que servirán de insumo.

#### **DISEÑO**

En esta etapa, el líder del proyecto comenzará con el diseño de un área de preparación, en la cuál se cargará la información proveniente de producción, posteriormente realizará un análisis de la información actual con el fin de identificar si cumple con los requisitos necesarios para ser utilizada, finalmente diseñará la estructura del datawarehouse tomando en cuenta los datos que este contendrá, su estructura y su contenido.

#### **CONSTRUCCIÓN**

Inicialmente se identifican todas las fuentes de datos y se extraen los mismos al área de preparación (Staging Area) esto corresponde a la etapa de Extracción.

Luego, los datos extraídos de las fuentes se compilan, convierten, reformatean y limpian en el área de preparación (Staging Area) para alimentar la base de datos de destino (Data Warehouse), esta es la etapa de Transformación y Limpieza.

Finalmente, los datos procedentes de la fase anterior (fase de Transformación y Limpieza) son cargados en el sistema de destino que formará nuestro Data Warehouse.

### **METODOLOGIAS**

Al diseñar una solución de Data Warehouse, las organizaciones se topan con distintas metodologías a seguir, cada una con sus ventajas. Así como ciertas características limitantes. Las cuales se deberán evaluar para seleccionar la que mejor se adapte a los requisitos del proyecto y la organización.

#### **RALPH KIMBALL - MULTIDIMENSIONAL**

La metodología multidimensional o metodología de Ralph Kimball, mantiene un diseño ascendente. Por lo que los Data Marts son los primeros en crearse y después se integran al Data Warehouse, donde se busca que el almacenamiento de datos de los usuarios se ejecute de la forma más rápida posible. Según Kimball, un Data Warehouse es la copia de los datos transaccionales específicamente estructurados para consultas analíticas e informes, con el fin de apoyar la toma de decisiones. Con esta metodología, al crear primero los Data Marts se proporcionan capacidades analíticas de informes para procesos específicos de negocio y funcionales.

**Principales Ventajas:**

* No requiere un equipo muy grande de desarrolladores y arquitectos de datos para mantener el Data Warehouse (menor costo).
* Brinda buena funcionalidad y seguimiento de las métricas o KPI´s, orienta los Data Marts a informes en cuanto a procesos de departamento o de negocios.
* Administración más “simplificada” al estar concentrado en los procesos y las áreas individuales en vez de toda la organización.
* La optimización de consultas es sencilla, predecible y controlable.

**Principales Desventajas:**

* Por su enfoque en procesos y áreas, puede no llegar a cubrir o manejar todos los requisitos en los informes.
* Consta de una menor flexibilidad de modificación.

Diagrama

Descripción generada automáticamente

#### **RALPH KIMBALL – RELACIONAL**

La metodología relacional de Bill Inmon muestra un diseño descendente, donde se construye primero el Data Warehouse y posteriormente los Data Marts. Ubicando el Data Warehouse en el centro de la información corporativa lo que asegura un marco lógico en los datos. Crea una estructura de entidades procurando que no se repitan datos. Este modelo crea una única fuente de verdad para todo el negocio. La carga de datos se vuelve menos compleja debido a la estructura normalizada del modelo. Sin embargo, el uso de esta disposición para realizar consultas es complicado; ya que incluye gran cantidad de tablas y vínculos. Este modelo propone la construcción de Data Marts por separado para cada departamento. Todos los datos que entran en el Data Warehouse están integrados para garantizar la integridad y la coherencia en toda la empresa. Actuando el Data Warehouse como el único origen de datos.

**Principales Ventajas:**

* El Data Warehouse proporciona una única versión de la verdad, al ser el único origen de datos para los Data Marts.
* Tiene una mayor facilidad de comprensión de los procesos empresariales para los usuarios, ya que el modelo lógico representa entidades empresariales detalladas.
* Resulta más fácil y menos propenso al fracaso el proceso de ETL, puesto que en la actualización de los datos y las anomalías se evitan al contar con una redundancia muy baja.
* Mayor flexibilidad a cambios por necesidades analíticas, de negocio y/o por fuentes de datos.

**Principales Desventajas:**

* De mayor complejidad, se requieren recursos con mayor capacidad en modelado y almacenamiento de datos (generalmente de mayor costo).
* Suele requerir de tiempos más largos para dado su procesos y entrega.

Diagrama

Descripción generada automáticamente

## **APARTADO PRACTICO**

### **AMBIENTE DE PRODUCCIÓN**

La base de datos Northwind es una base de datos de muestra creada originalmente por Microsoft y utilizada como base para sus tutoriales en una variedad de productos de bases de datos durante varios años. Ésta contiene los datos sobre las ventas de una empresa ficticia llamada "Northwind Traders", que importa y exporta alimentos especiales de todo el mundo. Esta base de datos fue replicada en SQL Server y fue llamada NorwhtindDB.

### **AREA DE PREPARACIÓN (STAGING AREA)**

Para la creación del Staging Area, (NW\_SA) se ejecutaron los siguientes scripts para la creación de sus tablas en SQL Server:

CREATE TABLE Categories(

CategoryID INT,

CategoryName VARCHAR(15),

Description VARCHAR(MAX),

Picture VARCHAR(MAX))

CREATE TABLE Customers(

CustomerID VARCHAR(5),

CompanyName VARCHAR(40),

ContactName VARCHAR(30),

ContactTitle VARCHAR(30),

Address VARCHAR(60),

City VARCHAR(15),

Region VARCHAR(15),

PostalCode VARCHAR(10),

Country VARCHAR(15),

Phone VARCHAR(24),

Fax VARCHAR(24))

CREATE TABLE Employees(

EmployeeID INT,

LastName VARCHAR(20),

FirstName VARCHAR(10),

Title VARCHAR(30),

TitleOfCourtesy VARCHAR(25),

BirthDate DATETIME ,

HireDate DATETIME ,

Address VARCHAR(60),

City VARCHAR(15),

Region VARCHAR(15),

PostalCode VARCHAR(10),

Country VARCHAR(15),

HomePhone VARCHAR(24),

Extension VARCHAR(4),

Photo VARCHAR(MAX),

Notes VARCHAR(MAX),

ReportsTo INT,

PhotoPath VARCHAR(255))

CREATE TABLE OrderDetails(

OrderID INT,

ProductID INT,

UnitPrice DECIMAL(25,2),

Quantity INT,

Discount DECIMAL(25,2))

CREATE TABLE Orders(

OrderID INT,

CustomerID VARCHAR(5),

EmployeeID INT,

OrderDate DATETIME ,

RequiredDate DATETIME ,

ShippedDate DATETIME ,

ShipVia INT,

Freight DECIMAL(25,2),

ShipName VARCHAR(40),

ShipAddress VARCHAR(60),

ShipCity VARCHAR(15),

ShipRegion VARCHAR(15),

ShipPostalCode VARCHAR(10),

ShipCountry VARCHAR(15))

CREATE TABLE Products(

ProductID INT,

ProductName VARCHAR(40),

SupplierID INT,

CategoryID INT,

QuantityPerUnit VARCHAR(20),

UnitPrice DECIMAL(25,2),

UnitsInStock INT,

UnitsOnOrder INT,

ReorderLevel INT,

Discontinued INT)

CREATE TABLE Shippers(

ShipperID INT,

CompanyName VARCHAR(40),

Phone VARCHAR(24))

CREATE TABLE Suppliers(

SupplierID INT,

CompanyName VARCHAR(40),

ContactName VARCHAR(30),

ContactTitle VARCHAR(30),

Address VARCHAR(60),

City VARCHAR(15),

Region VARCHAR(15),

PostalCode VARCHAR(10),

Country VARCHAR(15),

Phone VARCHAR(24),

Fax VARCHAR(24),

HomePage VARCHAR(MAX))

CREATE TABLE EmployeeTerritories(

EmployeeID INT,

TerritoryID VARCHAR(20))

CREATE TABLE Territories(

TerritoryID VARCHAR(20),

TerritoryDescription VARCHAR(50),

RegionID INT)

CREATE TABLE Region(

RegionID INT,

RegionDescription VARCHAR(50))

### **DATAWAREHOUSE**

Finalmente, se desarrolló la creación del Data Warehouse NW\_DW, aplicando un modelo de tipo estrella. La desnormalización se realizó de manera que las tablas en segundo nivel heredaran su información más importante a las tablas principales para conservar la estructura del modelo sin comprometer la integridad de los datos.

Se presenta los scripts para la creación de las tablas de dimensiones (DIM):

**-- Creación de la tabla DIM\_TIME, que contiene todos los días de 1996-1998**

CREATE TABLE DIM\_TIME(

ID\_TIME INT IDENTITY NOT NULL CONSTRAINT PK\_TIME PRIMARY KEY,

DATE DATE NULL,

YEAR INT NULL,

MONTH INT NULL,

DAY INT NULL,

NDAY VARCHAR(30) NULL,

NMONTH VARCHAR(20) NULL,

DAY\_YEAR INT NULL,

PERIOD VARCHAR(30) NULL)

**-- Reinicio de la tabla DIM\_TIME**

SELECT \* FROM DIM\_TIME

DELETE FROM DIM\_TIME

DBCC CHECKIDENT (DIM\_TIME, RESEED, 0)

**-- Procedimiento para llenar la tabla DIM\_TIME**

BEGIN

SET LANGUAGE English;

DECLARE @FECHA\_INICIO DATE;

DECLARE @FECHA\_FINAL DATE;

SET @FECHA\_INICIO='1996-01-01';

SET @FECHA\_FINAL ='1998-12-31';

DECLARE @PFECHA DATE;

DECLARE @PANNIO INT;

DECLARE @PMES INT;

DECLARE @PNMES VARCHAR(20);

DECLARE @PDIA INT;

DECLARE @PDIA\_ANNIO INT ;

DECLARE @PERIODO VARCHAR(30);

DECLARE @PNDIA VARCHAR(30);

WHILE @FECHA\_INICIO<=@FECHA\_FINAL

BEGIN

SET @PFECHA = @FECHA\_INICIO;

SET @PANNIO = YEAR(@FECHA\_INICIO);

SET @PMES = MONTH(@FECHA\_INICIO);

SET @PNMES = DATENAME(MONTH, @FECHA\_INICIO);

SET @PNDIA = DATENAME(dw,@FECHA\_INICIO);

SET @PDIA = DAY(@FECHA\_INICIO);

SET @PDIA\_ANNIO = DATENAME(DAYOFYEAR, @FECHA\_INICIO) ;

SET @PERIODO = CONCAT( CASE WHEN @PMES BETWEEN 1 and 9 THEN

CONCAT('0',CAST(@PMES AS VARCHAR(1))) ELSE

CAST( @PMES AS VARCHAR(2)) END ,' - ',DATENAME(MONTH, @FECHA\_INICIO))

INSERT INTO DIM\_TIME (DATE, YEAR, MONTH, DAY, NDAY, NMONTH, Y\_YEAR, PERIOD )

VALUES (@PFECHA, @PANNIO, @PMES, @PDIA, UPPER(@PNDIA), UPPER(@PNMES), @PDIA\_ANNIO, UPPER(@PERIODO)) SET @FECHA\_INICIO=DATEADD(DAY,1,@FECHA\_INICIO);

END ;

END ;

**-- Creación de la tabla DIM\_EMPLOYEES, que contiene los datos desnormalizados del empleado.**

**-- La columna TERRITORY adopta la region en la que trabaja cada empleado.**

CREATE TABLE DIM\_EMPLOYEES(

ID\_EMPLOYEE INT IDENTITY NOT NULL CONSTRAINT PK\_EMPLOYEE PRIMARY KEY,

EMPLOYEE\_ID INT NOT NULL,

EMPLOYEE\_NAME VARCHAR(80) NOT NULL,

TITLE VARCHAR(40) NOT NULL,

BIRTH\_DATE DATE NOT NULL,

HIRE\_DATE DATE NOT NULL,

CITY VARCHAR(20) NOT NULL,

REGION VARCHAR(20) NOT NULL,

COUNTRY VARCHAR(20) NOT NULL,

TERRITORY VARCHAR(20) NOT NULL,

REPORTS\_TO VARCHAR(60) NOT NULL)

**-- Creación de la tabla DIM\_ORDERS, que contiene los datos desnormalizados de las ordenes.**

**-- Incorpora la columna “SHIPPER”**

CREATE TABLE DIM\_ORDERS(

ID\_ORDER INT IDENTITY NOT NULL CONSTRAINT PK\_ORDER PRIMARY KEY,

ORDER\_ID INT NOT NULL,

CUSTOMERID VARCHAR(20) NOT NULL,

CUSTOMER VARCHAR(80) NOT NULL,

EMPLOYEE VARCHAR(80) NOT NULL,

ORDER\_DATE DATE NOT NULL,

REQUIRED\_DATE DATE NOT NULL,

SHIPPED\_DATE DATE NOT NULL,

SHIPPER VARCHAR(30) NOT NULL,

FREIGHT DECIMAL(25,2) NOT NULL,

SHIP\_CITY VARCHAR(30) NOT NULL,

SHIP\_REGION VARCHAR(20) NOT NULL,

SHIP\_COUNTRY VARCHAR(30) NOT NULL)

**-- Creación de la tabla DIM\_CUSTOMERS, que contiene los datos desnormalizados de los clientes.**

CREATE TABLE DIM\_CUSTOMERS(

ID\_CUSTOMER INT IDENTITY NOT NULL CONSTRAINT PK\_CUSTOMER PRIMARY KEY,

CUSTOMER\_ID VARCHAR(15) NOT NULL,

COMPANY\_NAME VARCHAR(80) NOT NULL,

CONTACT\_NAME VARCHAR(80) NOT NULL,

CONTACT\_TITLE VARCHAR(80) NOT NULL,

CITY VARCHAR(40) NOT NULL,

REGION VARCHAR(40) NOT NULL,

COUNTRY VARCHAR(40) NOT NULL,

CATEGORY VARCHAR(40) NOT NULL,

ENTRY\_DATE DATE NOT NULL)

**-- Creación de la tabla DIM\_PRODUCTS, con los datos desnormalizados de los productos .**

**-- Incorpora las columnas “Category”, “Description” y “Supplier”**

CREATE TABLE DIM\_PRODUCTS(

ID\_PRODUCT INT IDENTITY NOT NULL CONSTRAINT PK\_PRODUCT PRIMARY KEY,

PRODUCT\_ID INT NOT NULL,

PRODUCT\_NAME VARCHAR(80) NOT NULL,

CATEGORY VARCHAR(30) NOT NULL,

DESCRIPTION VARCHAR(100) NOT NULL,

SUPPLIER VARCHAR(100) NOT NULL,

REORDER\_LEVEL INT NOT NULL

)

**-- Creación de la tabla ORDER\_DETAILS, que se utiliza para facilitar el acceso a los datos.**

**-- En la práctica se utiliza un método distinto a esta tabla.**

CREATE TABLE ORDER\_DETAILS (

ORDERID INT NOT NULL,

PRODUCTID INT NOT NULL,

UNITPRICE DECIMAL(25,2) NOT NULL,

QUANTITY INT NOT NULL,

DISCOUNT DECIMAL(25,2) NOT NULL)

**-- Creación de la tabla FACT\_SALES, (Tabla de Hechos ) con los datos relevantes de cada orden.**

**-- Se relaciona con las demás tablas por medio de los IDs en cada una de ellas.**

CREATE TABLE FACT\_SALES(

ID\_SALE INT IDENTITY NOT NULL CONSTRAINT PK\_SALE PRIMARY KEY CLUSTERED,

ID\_EMPLOYEE INT NOT NULL,

ID\_ORDER INT NOT NULL,

ID\_CUSTOMER INT NOT NULL,

ID\_PRODUCT INT NOT NULL,

ID\_TIME INT NOT NULL,

QUANTITY INT NOT NULL,

UNIT\_PRICE DECIMAL(25,2) NOT NULL,

DISCOUNT DECIMAL(25,2) NOT NULL,

ORDER\_DATE DATE NOT NULL)

**-- Creación de la tabla METRICAS, utilizada en el análisis de datos con Power BI u otras herramientas.**

CREATE TABLE METRICAS(

IDENTIFICADOR INT NULL)

**-- Vinculación de llaves foráneas para el modelo estreda del Data Warehouse.**

ALTER TABLE FACT\_SALES WITH CHECK ADD CONSTRAINT FK\_SALES\_EMPLOYEES

FOREIGN KEY(ID\_EMPLOYEE) REFERENCES DIM\_EMPLOYEES (ID\_EMPLOYEE)

ALTER TABLE FACT\_SALES WITH CHECK ADD CONSTRAINT FK\_SALES\_ORDERS

FOREIGN KEY(ID\_ORDER) REFERENCES DIM\_ORDERS (ID\_ORDER)

ALTER TABLE FACT\_SALES WITH CHECK ADD CONSTRAINT FK\_SALES\_CUSTOMERS

FOREIGN KEY(ID\_CUSTOMER) REFERENCES DIM\_CUSTOMERS (ID\_CUSTOMER)

ALTER TABLE FACT\_SALES WITH CHECK ADD CONSTRAINT FK\_SALES\_PRODUCTS

FOREIGN KEY(ID\_PRODUCT) REFERENCES DIM\_PRODUCTS (ID\_PRODUCT)

ALTER TABLE FACT\_SALES WITH CHECK ADD CONSTRAINT FK\_SALES\_TIME

FOREIGN KEY(ID\_TIME) REFERENCES DIM\_TIME (ID\_TIME)

**POBLADO DE DATAWAREHOUSE (DIMENSIONES) CON PENTAHO**

La principal herramienta que hizo posible la creación del Data Warehouse es Pentaho, que posee características para la integración de datos de diferentes fuentes (archivos .CSV, tablas de Excel, diferentes tipos de bases de datos, entre otros).

Se crearon 3 conexiones a las bases de datos Northwind, NW\_SA (Northwind Stagin Area) y NW\_DW (Northwind Data Warehouse) para lograr establecer la comunicación necesaria entre ellas para la transferencia de los datos. A continuación se presentan los querys utilizados para obtener la información desde NW\_SA hacia NW\_DW:

**-- Poblado de la tabla DIM\_EMPLOYEES**

SELECT DISTINCT

UPPER(E.EMPLOYEEID) EMPLOYEE\_ID,

UPPER(CONCAT(E.FIRSTNAME, ' ', E.LASTNAME)) EMPLOYEE\_NAME,

UPPER(E.TITLE) TITLE,

CAST(E.BIRTHDATE AS DATE) BIRTH\_DATE,

CAST(E.HIREDATE AS DATE) HIRE\_DATE,

UPPER(E.CITY) CITY,

UPPER(isnull(E.REGION,'NO REGION')) REGION,

UPPER(E.COUNTRY) COUNTRY,

UPPER(R.REGIONDESCRIPTION )TERRITORY,

COALESCE(NULLIF(

UPPER(CONCAT(J.FIRSTNAME, ' ', J.LASTNAME)),''), 'NO BOSS') REPORTS\_TO

FROM EMPLOYEES E

LEFT JOIN EMPLOYEES J

ON E.REPORTSTO = J.EMPLOYEEID

LEFT JOIN EMPLOYEETERRITORIES ET

ON E.EMPLOYEEID = ET.EMPLOYEEID

LEFT JOIN TERRITORIES T

ON ET.TERRITORYID = T.TERRITORYID

LEFT JOIN REGION R

ON T.REGIONID = R. REGIONID

ORDER BY EMPLOYEE\_ID

**-- Poblado de tabla DIM\_CUSTOMERS:**

SELECT

UPPER(C.CUSTOMERID) CUSTOMER\_ID,

UPPER(C.COMPANYNAME) COMPANY\_NAME,

UPPER(C.CONTACTNAME) CONTACT\_NAME,

UPPER(C.CONTACTTITLE) CONTACT\_TITLE,

UPPER(C.CITY) CITY,

UPPER(ISNULL(C.REGION,'NO REGION')) REGION,

UPPER(C.COUNTRY) COUNTRY,

ISNULL(

CASE WHEN cc.grupo=1 THEN

'A'

WHEN CC.GRUPO=2 THEN

'B'

WHEN CC.GRUPO=3 THEN

'C'

WHEN CC.GRUPO=4 THEN

'D'

END ,'E' ) CATEGORY,

ISNULL(CAST(IC.ENTRY AS DATE),CAST(GETDATE() AS DATE)) ENTRY\_DATE

FROM CUSTOMERS C

LEFT JOIN (

SELECT MIN(ORDERDATE) ENTRY, CUSTOMERID FROM ORDERS

GROUP BY CUSTOMERID

) IC

ON C.CUSTOMERID=IC.CUSTOMERID

LEFT JOIN (

SELECT CUSTOMERID, SUM(QUANTITY \* UNITPRICE) SALES,

NTILE(4) OVER(ORDER BY SUM(QUANTITY \* UNITPRICE) DESC) AS GRUPO

FROM ORDERS O

INNER JOIN ORDERDETAILS OD

ON O.ORDERID = OD.ORDERID

WHERE SHIPPEDDATE IS NOT NULL

GROUP BY CUSTOMERID) CC

ON C.CUSTOMERID = CC.CUSTOMERID

**-- Poblado de la tabla DIM\_ORDERS**

SELECT

UPPER(O.ORDERID) ORDER\_ID,

UPPER(C.CUSTOMERID) CUSTOMERID,

UPPER(C.COMPANYNAME) CUSTOMER,

UPPER(CONCAT(E.FIRSTNAME, ' ', E.LASTNAME)) EMPLOYEE,

CAST(O.ORDERDATE AS DATE) ORDER\_DATE,

CAST(O.REQUIREDDATE AS DATE) REQUIRED\_DATE,

isnull(CAST(O.SHIPPEDDATE AS DATE),CAST(getDATE() AS DATE)) SHIPPED\_DATE,

UPPER(S.COMPANYNAME) SHIPPER,

UPPER(O.FREIGHT) FREIGHT,

UPPER(O.SHIPCITY) SHIP\_CITY,

UPPER(isnull(O.SHIPREGION,'NO SHIP REGION')) SHIP\_REGION,

UPPER(O.SHIPCOUNTRY) SHIP\_COUNTRY

FROM ORDERS O

INNER JOIN EMPLOYEES E

ON E.EMPLOYEEID = O.EMPLOYEEID

INNER JOIN CUSTOMERS C

ON C.CUSTOMERID = O.CUSTOMERID

INNER JOIN SHIPPERS S

ON S.SHIPPERID = O.SHIPVIA

ORDER BY ORDER\_ID

**-- Poblado de tabla FACT\_SALES:**

SELECT

E.ID\_EMPLOYEE,

O.ID\_ORDER,

C.ID\_CUSTOMER,

PR.ID\_PRODUCT,

TI.ID\_TIME,

OD.QUANTITY,

OD.UNITPRICE UNIT\_PRICE,

OD.DISCOUNT,

O.ORDER\_DATE

FROM ORDER\_DETAILS OD

INNER JOIN DIM\_ORDERS O

ON OD.ORDERID = O.ORDER\_ID

INNER JOIN DIM\_PRODUCTS PR

ON OD.PRODUCTID = PR.PRODUCT\_ID

INNER JOIN DIM\_TIME TI

ON TI.DATE = O.ORDER\_DATE

INNER JOIN DIM\_CUSTOMERS C

ON C.COMPANY\_NAME = O.CUSTOMER

INNER JOIN DIM\_EMPLOYEES E

ON E.EMPLOYEE\_NAME = O.EMPLOYEE

**-- Poblado de tabla ORDER\_DETAILS:**

SELECT \* FROM ORDERDETAILS

**-- Poblado de tabla DIM\_PRODUCTS**

SELECT

UPPER(P.PRODUCTID) PRODUCT\_ID,

UPPER(P.PRODUCTNAME) PRODUCT\_NAME,

UPPER(C.CATEGORYNAME) CATEGORY,

UPPER(C.DESCRIPTION) DESCRIPTION,

UPPER(S.COMPANYNAME) SUPPLIER,

UPPER(P.REORDERLEVEL) REORDER\_LEVEL

FROM PRODUCTS P

INNER JOIN SUPPLIERS S

ON S.SUPPLIERID = P.SUPPLIERID

INNER JOIN CATEGORIES C

ON C.CATEGORYID = P.CATEGORYID

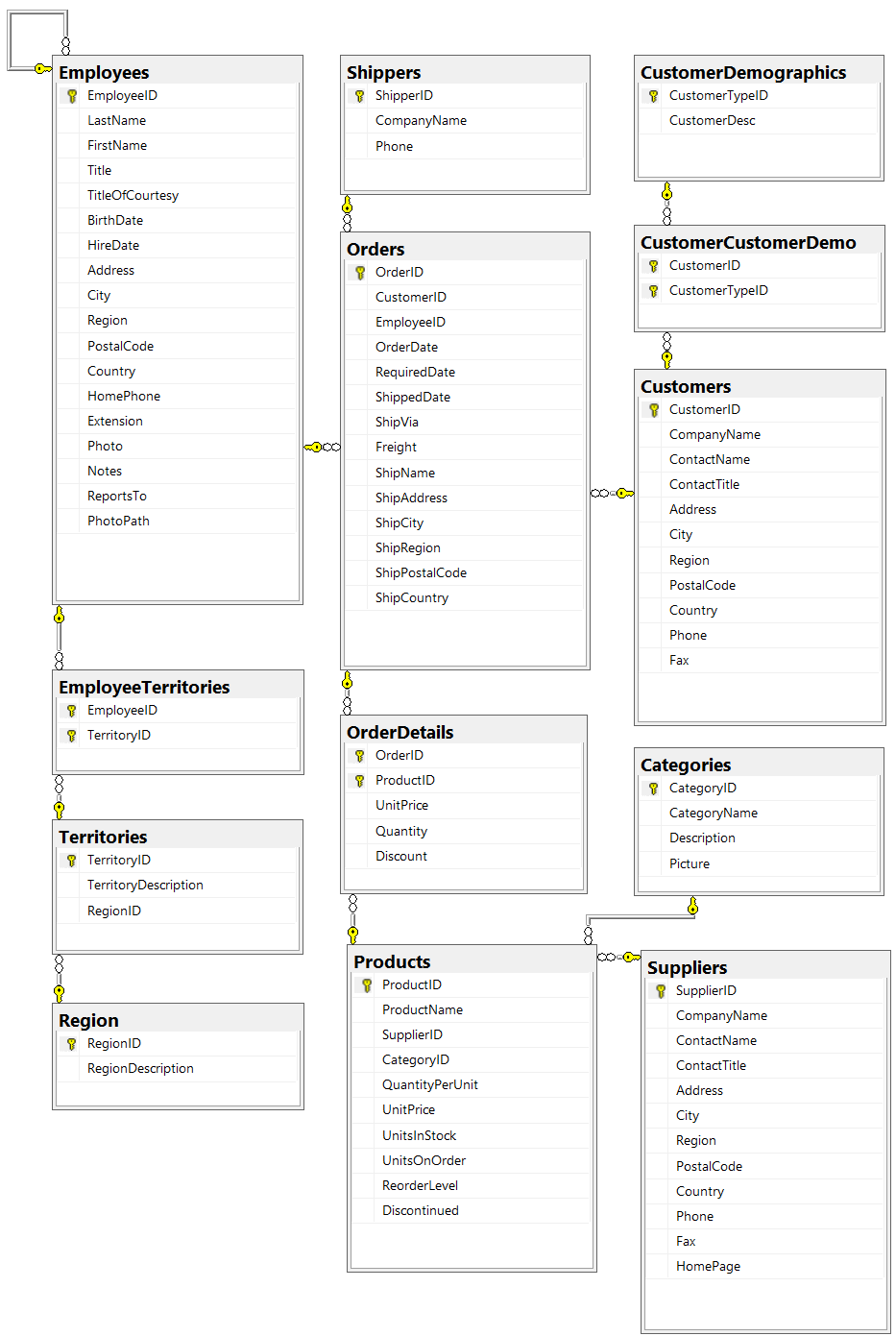
**DIAGRAMAS**

Diagrama Base de Datos Northwind (Producción).

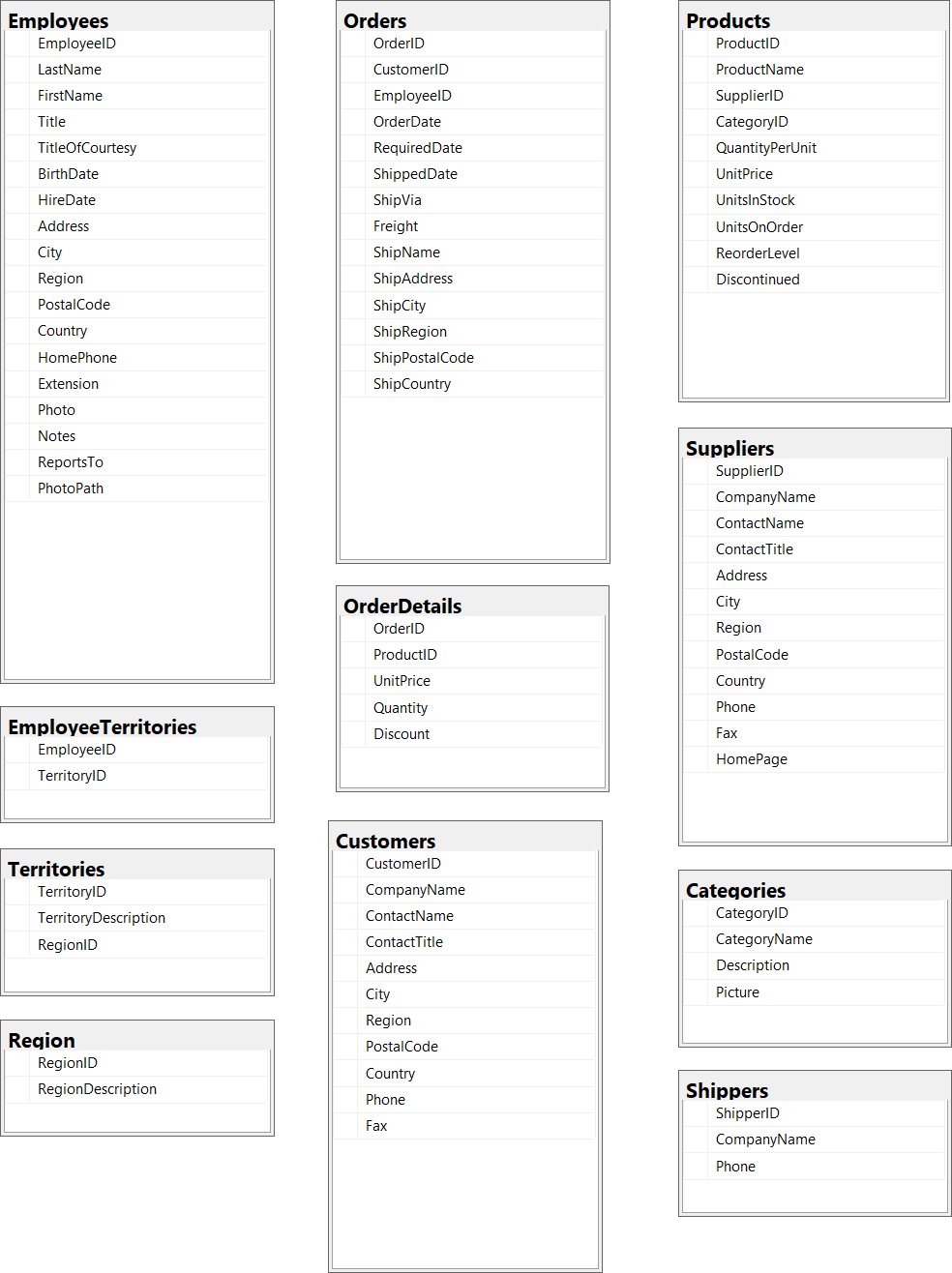


Diagrama Base de datos NW\_SA (Northwind Staging Area).

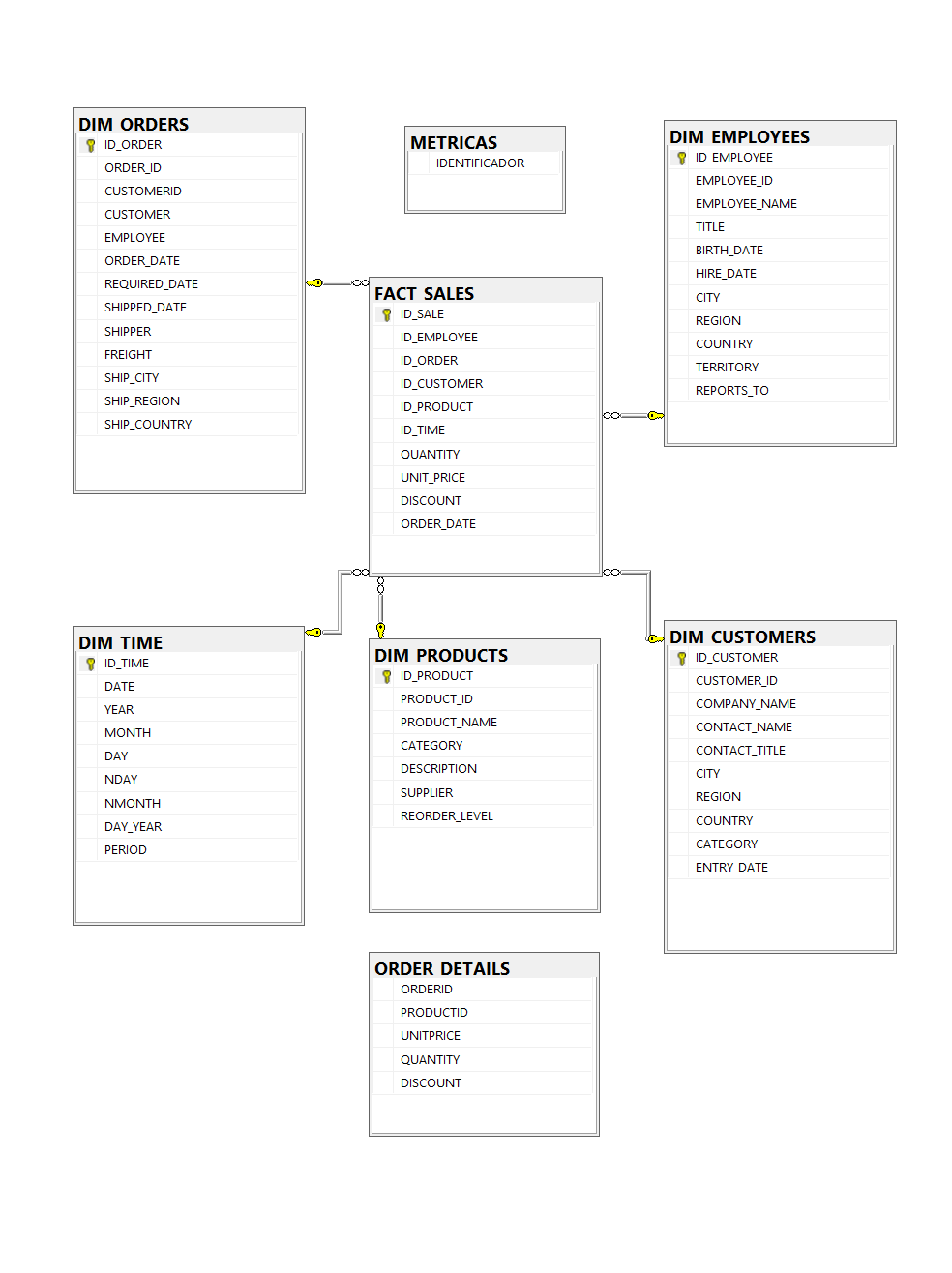


Diagrama Base de Datos NW\_DW (Northwind Data Warehouse).

**CONCLUSIÓN**

El desarrollo de un datawarehouse es un proceso técnico sumamente complejo pero que se ve apoyado de manera importante de un área de negocio mediante el establecimiento de los objetivos y la dirección de tal manera que se convierte en una relación bidireccional.

Para esto existen diversas técnicas, herramientas, tecnologías que pueden ser utilizadas y no existe una perfecta o única; por el contrario, estas deben ser elegidas cuidadosamente según las necesidades del proyecto o de la organización.

Durante la ejecución de este proceso, es imperativo realizar un análisis de la estructura que posee actualmente la organización con el fin de estandarizarla mediante técnicas de ETL para asegurar el cumplimiento de los objetivos.

# **BIBLIOGRAFÍA**

Fatima, N. (2022, 24 noviembre). *Proceso ETL y los pasos para su implementación*. Astera. Recuperado 12 de marzo de 2023, de <https://www.astera.com/es/type/blog/etl-process-and-steps/>

Alfaro, F. R. (2021, 21 junio). *Proceso ETL con SQL Server Integration Services - SSIS*. Data Management. Recuperado 12 de marzo de 2023, de <https://datamanagement.es/2020/04/06/proceso-etl-con-sql-server-integration-services-ssis/>

Mendoza A. (2022, 23 noviembre). *MetodologíasData Warehouse | Gravitar Business Intelligence*. Business Intelligence, Data Warehouse, Monterrey, México : Gravitar. Recuperado 12 de marzo de 2023, de <https://gravitar.biz/datawarehouse/metodologias-data-warehouse/>

Ramirez J. (2022, 10 agosto). Recuperado 12 de marzo de 2023, de <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/qu-son-los-procesos-etl>