UNIVERSITY OF ECONOMICS AND LAW

FACULTY OF INFORMATION SYSTEM

FINAL PROJECT

TOPIC:Data Warehouse In Wide World Importers

Subject: Data Warehouse and Integration

Lecturer: Le Ba Thien

Ho Chi Minh, November 29th, 2023

Members of Group

No.	Student ID	Full name	Leader	Performance
1		Nguyễn Thành Luân		100%
2				

3				
4				

Acknowledgments

First and foremost, we would like to express our sincere gratitude to the University of Economics and Law for incorporating Data Warehousing and Integration into the curriculum. In particular, we want to extend our deep appreciation to our research advisor and course lecturer - Mr. Le Ba Thien, Mr. Nguyen Quang Nhat, for their dedicated guidance and imparting invaluable knowledge to us throughout the recent study period. During our participation in the Data Warehouse class, we have accumulated additional useful knowledge, enhanced analytical abilities, and fostered a spirit of serious and effective learning. Their enthusiasm and dedication have been a significant source of motivation that has helped us complete this essay thoroughly. Without their encouragement and guidance, this essay could not have been executed as efficiently and meticulously as it has.

The Data Warehousing and Integration course is an interesting and extremely useful subject with a high application of specialized knowledge, ensuring sufficient relevance to students' professional disciplines. We have endeavored to complete this essay to the best of our abilities, utilizing textbook knowledge and the experience we have accumulated. However, due to limitations in knowledge capital and the ability to gather real-world data, there are still many surprises. Despite our best efforts, it is certain that the essay may have shortcomings and incomplete aspects. Therefore, our group looks forward to receiving suggestions and sharing from teachers so that the group's plan can become more refined.

We wish everyone good health and much success in their teaching careers. Thank you!

Group LK2T

Commitment

We would like to assure you that the results presented below entirely reflect the application of our knowledge based on the teachings from the Department of Data Warehousing and Integration, Database, combined with reference materials from books, newspapers, and other media. We commit that this project does not plagiarize or copy any information from external sources. The author pledges that the project will be completed by December 5, 2023, and will be supervised by the instructor, Mr. Le Ba Thien.

Table of Content

List of Tables

List of Figures

Abstract

Our project focuses on constructing a Data Warehouse (DW) from the Sales module of the WideWorldImporters dataset, utilizing tools and technologies such as SQL Server, SQL Server Integration Services (SSIS), and Power BI. The primary objective of the project is to transform data from various sources within the Sales module of the WideWorldImporters dataset into a high-quality DW, ready for analysis and reporting.

Using SQL Server and SSIS, we efficiently performed the Extract, Transform, Load (ETL) process, ensuring data consistency and reliability within the DW. Once the DW was built and data was loaded, we employed Power BI to create a diverse set of dashboards for visualizing information and supporting business decision-making.

The ultimate outcome of the project is a robust DW system and a series of user-friendly dashboards, making data analysis more accessible and effective. This DW and its dashboards will play a crucial role in aiding business decision-making and optimizing the performance in the future.

Chapter 1. Project Overview

1.1 Business case

In today's dynamic business environment, the integration and effective utilization of data across various departments is crucial for organizational success. The WideWorldImporters Data Warehouse project is designed to align with this necessity, concentrating on the integration of customer and product analytics within the sales sector. This business case aims to articulate the rationale behind the project, emphasizing the anticipated advantages, financial considerations, and the expected impact on the overall business operations and strategic planning.

The project's core objective is to foster a cohesive operational framework, eliminating isolated departmental efforts and promoting a synchronized decision-making process. By analyzing, developing, and rigorously testing a solution that spans across different departments, the project seeks to leverage data warehousing and integration techniques. This approach not only facilitates efficient data storage but also addresses specific needs in Purchasing and Production. The result is an enterprise-wide solution that not only streamlines data processing but also enhances the accuracy and effectiveness of business decisions, reflecting a holistic view of the company's operations.

Our detailed Business Case: Data Warehouse for Customer and Product Analysis in Sales. This project is dedicated to constructing a data warehouse that focuses on the Sales module of the WideWorldImporters database. The primary aim is to leverage this data warehouse for in-depth analysis of customer behaviors and product performance. By doing so, we intend to optimize our sales strategies and improve overall business efficiency. Beside that, the core mission of this project is to extract meaningful insights related to customer interactions and product dynamics within the sales framework of WideWorldImporters. Our objectives are centered around resolving two key aspects:

+ In-depth Customer Analysis: Understanding customer purchasing patterns, preferences, and behaviors. Segmenting customers based on various criteria to tailor marketing and sales strategies effectively.
+ Comprehensive Product Performance Evaluation: Assessing product sales trends, profitability, and market demand. Identifying factors contributing to the success or underperformance of products in different market segments.

In conclusion, this project is set to transform the sales methodology of WideWorldImporters through advanced data analytics. By establishing a dedicated

data warehouse for the Sales module, the project lays the foundation for a more dynamic, informed, and customer-focused business strategy. This initiative is poised to make a

substantial impact on the company's ongoing growth and competitive edge.

1.2 Objectives

The project is undertaken with the primary goal of answering critical business questions that revolve around optimizing sales strategies and understanding customer preferences in depth. To achieve this, we have formulated our objectives into five key questions, each aiming to unravel specific aspects of sales and customer engagement:

(1) When in the Year Brings the Highest Revenue?
This question aims to pinpoint the period in the year that generates the highest revenue from sales data. This evaluation assists in understanding and observing business operation cycles, thereby aiding in planning and optimizing business strategies.

(2) What Are the Best-Selling Products and Where Are They Primarily Concentrated?
This question targets identifying the best-selling products along with their primary consumption locations. Understanding this assists in refining distribution strategies and marketing efforts to maximize sales figures.

(3) Where Are the Most Profitable Products Concentrated?
Answering this question helps understand which products yield the highest profits and where they are predominantly sold. Insight into this aspect aids in decisions regarding pricing strategies, inventory management, and distribution channel management.

(4) Statistics on Purchase Frequency of Customer Groups per Year The objective of this statistical analysis is to examine the purchasing patterns of customer groups per year. This information provides deeper insights into shopping preferences, changes in purchasing behavior over different periods, enabling fine-tuning of customer interaction strategies.

(5) Which Supplier's Products Are the Best Sellers?
This question aims to identify the best-selling products from a specific supplier. Understanding this information contributes to evaluating the supplier relationship, determining the impact of these products, and optimizing supplier partnership management.

These objectives, framed as specific questions, will direct our analytical efforts and ensure that the outcomes of this project are directly aligned with enhancing business performance, understanding customer behavior, and improving product strategy within

the WideWorldImporters framework.

1.3 Objects and scopes

1.3.1 Objects

This project primarily focuses on two key objects:

1. Customer Data: This encompasses all information related to customers in the WideWorldImporters database, including personal details, purchase history, and customer categorization (CustomerCategory).
2. Product Data: This includes detailed information about products, such as product types (StockGroupName and StockItemCategory), sales performance data, and purchasing trends associated with each product.

1.3.2 Scopes

The scope of the project is defined to ensure a comprehensive analysis with the following elements:

1. Sales Data Analysis: Focusing on transaction efficiency and customer behavior patterns using WideWorldImporters' sales data.
2. Comprehensive Analysis for Business Strategy: Determine business performance through products, customers in terms of time and location

1.4 Expected output of the project

The project is designed to yield a comprehensive set of outputs that will directly inform and enhance the business strategies of WideWorldImporters. Outputs are expected to include several customer and product rating metrics.

1.5 Tools and Programing language

For the successful execution of the WideWorldImporters sales data analysis project, we will be utilizing a combination of robust tools and programming languages, each chosen for their specific capabilities in handling, processing, and visualizing large datasets. The key tools and technologies to be used are:

- Microsoft SQL Server:As the primary database platform, Microsoft SQL Server will be used for storing and managing the WideWorldImporters dataset. It offers advanced data processing capabilities, security, and scalability. SQL (Structured

Query Language) will be the primary programming language for querying and manipulating the data within SQL Server. It will be used for writing scripts to extract, transform, and load (ETL) data, as well as for complex data analysis and reporting

tasks.

- SQL Server Integration Services (SSIS): SSIS, a component of Microsoft SQL Server, will be employed for the ETL processes. It is instrumental in integrating and consolidating data from different sources into the data warehouse. SSIS will be used to automate the data pipeline, ensuring efficient and error-free data loading and transformation. This includes data cleansing, normalization, and preparation for analysis.
- Microsoft Server Analysis Service (SSAS): This tool will be used by us in projects to create and manage multidimensional data cubes, providing the ability to analyze, make statistics and create detailed reports from data. Does it have different structures? This helps in analyzing and presenting complex information in a more intuitive and understandable way.
- Microsoft Power BI:For data visualization and business intelligence, Microsoft Power BI will be utilized. It is a powerful tool for creating interactive reports and dashboards, enabling easy interpretation of complex data. Power BI will facilitate the development of dynamic visualizations, which can provide insights into customer behaviors, product performance, and sales trends. These visualizations will be crucial for communicating findings to stakeholders and aiding in decision-making processes.

These tools, in combination, provide a comprehensive environment for data warehousing, analysis, and reporting. The use of SQL Server ensures robust data management, SSIS enables efficient data integration and preparation, and Power BI provides advanced capabilities for data visualization and business intelligence. This integrated approach will ensure the project delivers accurate, insightful, and actionable outputs.

1.6 Structure of project

Chapter 1: Introduction

This chapter provides the project overview, ranges of researching and outcomes.

Chapter 2: Related Research and Theoretical Basis

This chapter demonstrates the theoretical basis that are closely related to the topic of the project, including Data Warehouse, Business Intelligence,... and previous research related to the project topic

Chapter 3: Business requirements analysis

This chapter describes and analyzes the business requirements and the need to transform data-driven decisions by building and implementing a data warehouse.

Chapter 4: Building Data Warehouse and Integrating Data

This chapter indicates the complete data selection, table and schema design for data warehouse and details.

Chapter 5: Analysis and Recommendations for Businesses
In this section, we present and analyze the data, then draw important conclusions from the study. Specific recommendations are proposed based on these insights, with the goal of helping businesses improve operations and face challenges in their sector.

Chapter 2. Related Research and Theoretical Basis

2.1 Related Research

The article "A Data Warehouse-Based System for Service Customization

Recommendations in Product-Service Systems" was written by Laila Esheiba, Iman M. A. Helal, Amal Elgammal and Mohamed E. El-Sharkawi. It was published in the journal Sensors in 2022. This paper introduces a data warehouse-based system for service customization in a product-service system (PSS). PSS focuses on providing products with services, and PSS customization involves configuring the product to meet customer needs. In the article, the author proposes a recommendation system based on a data warehouse that collects and analyzes product usage data from similar products. Analyzing this data helps identify the most important components and the causes of problems. Based on this analysis, the system suggests the appropriate sensor type for the customer. The article also conducts an experimental study on a CNC milling machine to demonstrate the applicability of the suggestion system.

The article titled "Designing a Data Warehouse System for Sales and Distribution Company" was written by Balasingham Ragulan and Romal Subash. The article was published in the February 2021 issue of "Big Data Mining and Analytics". This article presents the design of a data warehouse system for a sales and distribution company. The need to move from manual data manipulation to digital data manipulation is necessary to survive in the industry, achieve business goals, and reuse historical data in the future. The article presents a data warehouse design model to build a data warehouse system for sales and distribution companies. This model includes data migration from existing sources, ETL process and data indexing and loading. The article also covers the architecture of data warehouse design and the process of cleaning and transforming data.

2.2 Theoretical Basis

Data Warehouse

Data Warehouse is a data storage system designed to support analysis and decision making in business environments. This is where data from many different sources is aggregated, stored, and maintained to create a comprehensive and analyzable source of information. Data Warehouse, as a strategic information storage center, sets out the task of converting data into knowledge

Important features of Data Warehouse include:

+ Data Aggregation: Integrate data from different sources such as online transaction databases (OLTP) and other systems to create a comprehensive data set. + Data Transformation: Apply ETL (Extraction, Transformation, Loading) process to extract, transform and load data into Data Warehouse.
+ Implementation Time: Data can be updated periodically or in real time to ensure modernity and accuracy.
+ Analytical Support: Provides a data structure that is easy to understand and

optimized for performing queries and analyzing data.

+ Query Capabilities: Enables users to perform complex queries and multidimensional analysis to make strategic decisions.

SQL(Structured Query Language)

SQL stands for "Structured Query Language". This is a programming language used to manage and interact with relational database management systems (RDBMS - Relational Database Management System). SQL provides commands and syntax to query, update, and manage data in databases. SQL can be used in many different database management systems, including Microsoft SQL Server, MySQL, PostgreSQL, Oracle, and many other RDBMS systems. Each system has some unique characteristics, but SQL provides a basic, standard set of commands that can be widely applied in many database environments.

ETL(Extract, Transform, and Load)

ETL is an acronym for Extract, Transform, and Load. This is an important process in the field of data processing and data warehouse. Below is a detailed description of each part of the ETL process.

+ Extract: This step involves collecting data from different sources, such as databases, pipelines, APIs, and others. Data is often extracted from production systems and stored temporarily in preparation for the next step.
+ Transform: Once the data has been extracted, it needs to be transformed to ensure that it fits the data model of the data warehouse or other data storage system. Business rules and standards are applied in this step to make the data understandable and suitable for analysis.
+ Load: After data has been extracted and transformed, it is loaded into the data warehouse or other data storage system to make it ready for querying and analysis. This step also includes error handling and data loading monitoring.

The ETL process helps organizations organize and prepare data from various sources so that it can be used effectively in analytics and reporting processes.

SSIS(SQL Server Integration Services)

SSIS stands for SQL Server Integration Services – a data management and integration tool that comes with Microsoft SQL Server databases. SSIS makes data storage more flexible and faster with features such as an ETL tool (extract large amounts of data from multiple data sources in different formats, edit and clean the data then load them into the designed data warehouse). Companies today use SSIS for data storage, data extraction and loading, data management, data cleaning, consolidation, and more. Before SSIS, moving data between storage locations was often difficult because the structure of the

data was different and it needed to be changed before moving to the new storage location. However, with SSIS, it can extract data from many different sources such as SQL Server databases, Excel files, Oracle and DB2 databases, etc. to deliver to the destination easily. In addition, SSIS also includes graphical tools and wizards to perform workflow functions such as sending email messages, file transfer operations, extracting data sources, and transferring data to destination.

OLTP(Online Transaction Processing)

OLTP stands for "Online Transaction Processing". This is a type of database system designed to process and manage online transactions in real time. These transactions typically involve adding, modifying, and querying data, and are frequently performed in everyday enterprise environments.

SCD(Slowly Changing Dimension)

SCD is an acronym for "Slowly Changing Dimension," which is a concept in the field of data management and data warehousing. Slowly Changing Dimensions is concerned with how to manage and track changes in data over time, especially when information in the data warehouse's dimension tables change.

There are three main types of Slowly Changing Dimensions (SCD): + SCD Type 1 (SCD1): In this method, when data changes, the old value is simply overwritten with the new value. Does not retain change history. This is the simplest method, but does not retain change history.

+ SCD Type 2 (SCD2): This method retains change history by creating new records for each change in data. Each new record has a unique primary key and an effective time, so that it can track how long a particular attribute's value is valid. This helps maintain track of past changes.
+ SCD Type 3 (SCD3): This method retains some history, but does not retain all information about the change. Instead, it retains some limited information about the change and manages a number of columns to hold the old and new values of the attribute.

Full Load + Incremental Load

Full Load and Incremental Load are two methods used in the process of updating data into the data warehouse. Below is a description for each method:

Full Load: Full Load is the process of loading all data from the original data source into the data warehouse. During this process, all data that already exists in the data warehouse will be deleted and replaced with the latest data from the source. The Full Load process is often performed initially when building a data warehouse or when there are changes to data structures or business rules. While Full Load ensures that the data in the data warehouse is complete and accurate, it can be time- and resource-intensive when

processing large data sets.

Incremental Load: Incremental Load is the process of loading only new or changed records from the data source into the data warehouse. Instead of reloading the entire data, only the records that are newest or have changed since the last load will be updated to the data warehouse. The Incremental Load process is usually performed periodically (e.g. daily, weekly) to ensure that the data warehouse is always updated with the latest data without reprocessing the entire data. This saves time and resources compared to Full Load.

Depending on the specific requirements and situations, a data warehouse system can use both Full Load and Incremental Load methods to ensure data is updated and complete in the data warehouse.

OLAP(Online Analytical Processing)

OLAP (Online Analytical Processing) is a data processing technology and methodology used in analyzing, querying, and reporting data in business analytics environments. OLAP focuses on handling data analysis activities from many different perspectives to provide insights and overviews of data. OLAP technology allows users to perform complex queries and analyze data from many different dimensions and measures. Dimensions represent aspects or attributes of data (e.g., time, location, product), while measures represent numerical data or quantitative metrics (e.g., revenue, sales volume). By combining dimensions and measures, users can investigate, analyze and find information from different perspectives. OLAP technology is often applied in data warehouse systems and business analytics systems to provide intelligent decision and management support based on integrated data and insightful analysis.

Star Schema

Star Schema is a popular database design model in data warehouse systems. It is used to organize data simply and effectively to support querying and data analysis. The Star Schema model includes two main components: Facts table and Dimensions table.

Facts table: The Facts table contains quantitative information or quantitative data (measures) that we want to analyze and query. For example, in a retail system, measures could be revenue, number of products sold, or total order value. Facts tables often contain data columns and foreign keys to link with Dimensions tables.

Dimensions table: The Dimensions table contains information that describes aspects or attributes of data (for example, time, location, product). Each Dimensions table usually contains a primary key column and columns describing detailed information about that aspect. For example, in the Time Dimensions table, columns might include day, month, year, quarter, and season.

The structure of the Star Schema model resembles a star, in which the Facts table is in the center and the Dimensions tables are located around it. The Facts table links with

Dimensions tables through foreign keys, allowing queries and analysis of data from different perspectives.

MDX

MDX (Multidimensional Expressions) is a widely used query language in the data warehouse environment and multidimensional data management systems such as Microsoft SQL Server Analysis Services (SSAS), Oracle OLAP, and various other OLAP systems. It is specifically designed for querying and interacting with data organized in a multidimensional model, often found in data cubes within OLAP systems. This model typically includes data cubes, dimensions, and various components like measures, all contributing to data analysis.

Kimball's Principle

According to Kimball's "bottom-up" principle, the process of building a data warehouse involves approaching each part individually, focusing on addressing specific business needs. In our research case, the goal is to identify the most frequent customer profiles, and assess the best-selling products from a large dataset comprising information about products and customers. By gradually integrating these components, the data warehouse becomes flexible, efficient, and scalable. This approach helps reduce risks and enhances the ability to quickly respond to complex business requirements. In summary, following Kimball's principle in constructing a data warehouse supports detailed analysis and provides essential information about order value, customer profiles, and product sales from a diverse and extensive dataset.

Chapter 3. Business requirements analysis

3.1 Introducing the Business

Wide World Importers, or WWI, is a wholesale importer and distributor of novelty goods based in the San Francisco bay area. They mostly serve other companies that sell to individuals. Their customers include all sorts of retail places across the US like specialty stores, supermarkets, computer shops, tourist spots, and even some folks directly. WWI also works with other wholesalers through agents who help market their products. While their current customers are all in the US, they're thinking about expanding to other places.

WWI buys goods from various suppliers—think novelty and toy makers, as well as other wholesalers. They store these items in their warehouse and reorder when needed to fill customer orders. They also sell packaging materials in smaller amounts for the customers' convenience after buying them in bulk.

3.2 Business requirements

In the dynamic world of wholesale importing and distribution, WWI sets forth several pivotal business requirements aimed at enhancing their operations and strategic decision-making. Each requirement is driven by a specific goal, aligning with the company's overarching mission to optimize revenue, identify key products and profit centers, and understand customer behavior.

The first requirement, *seeking the highest total revenue within a span of six months*, embodies WWI's quest for financial success and growth. This objective underscores the significance of maximizing revenue generation in a defined timeframe. By monitoring and analyzing revenue streams over this period, the company aims to identify peak performance periods, allowing them to replicate successful strategies and capitalize on market opportunities.

Identifying the best-selling products holds paramount importance in the company's strategy. Understanding the top-selling items provides invaluable insights into consumer preferences and market trends. Knowing which products resonate most with customers enables WWI to allocate resources effectively, potentially increasing stock levels of high-demand items while evaluating the need for less popular goods. *Additionally, pinpointing the primary geographical regions where these popular*

products sell assists in refining distribution strategies and targeting specific markets for further expansion.

Profitability is a core focus for WWI, as evident in their quest to determine the highest-profit products and their associated regions. This requirement emphasizes the company's dedication to maximizing returns on investment. *By assessing which products yield the most substantial profits and where they perform best geographically,* WWI can streamline procurement, inventory management, and marketing efforts, ensuring a strategic allocation of resources to bolster profitability.

Understanding customer behavior is crucial for any business, and WWI recognizes this by aiming to *analyze the purchasing frequency of customer groups annually.* This requirement serves the purpose of categorizing customer segments based on their buying habits. By examining purchasing patterns, the company gains valuable insights into customer preferences, enabling them to tailor marketing strategies, refine product

offerings, and enhance customer satisfaction.

Lastly, *identifying the best-selling products by supplier elucidates the strength of partnerships and the influence of supplier of erings on WWI's sales.* Recognizing which suppliers' products consistently perform well allows the company to strengthen relationships with these suppliers, potentially negotiating better terms or exploring opportunities for collaborative ventures to further drive sales and mutual growth.

3.3 Data Description

Microsoft launched the "Wide World Importers" sample data set in 2016. This is a sample data set released by Microsoft to support users in testing, training, and developing software solutions, especially is in the field of database management and data analysis. The WideWorldImporters dataset contains all transaction information and daily data for sales and purchases. It is divided into 4 main modules: Application, Purchasing, Sales and Warehouse.

Table 3.1 - 4 main modules of WWI dataset

Module	Description
Application	Application-wide users, contacts, and parameters. This schema also contains reference tables with data that is used by multiple schemas
Purchasing	Stock item purchases from suppliers and details about suppliers.
Sales	Stock item sales to retail customers, and details about customers and sales people.
Warehouse	Stock item inventory and transactions.

For the purpose of this article, we will only need the data from the Sales, Warehouse, and Purchasing modules to be enough for data warehouse construction and analysis.

Table 3.2 - Sales module

Table	Description
Customers	Main entity tables for customers (organizations or individuals)
CustomerCategories	Categories for customers (for example, novelty stores, supermarkets, etc.)
BuyingGroups	Customer organizations can be part of groups that exert greater buying power
CustomerTransactions	All financial transactions that are customer-related (invoices, payments)
SpecialDeals	Special pricing. This can include fixed prices, discount in dollars or discount percent.
Orders	Detail of customer orders
OrderLines	Detail lines from customer orders
Invoices	Details of customer invoices
InvoiceLines	Detail lines from customer invoices

Table 3.3 Warehouse module

Table	Description
StockItems	Main entity table for stock items
StockItemHoldings	Non-temporal columns for stock items. These are frequently updated columns.

StockGroups	Groups for categorizing stock items (for example, novelties, toys, edible novelties, etc.)
StockItemStockGroups	Which stock items are in which stock groups (many to many)
Colors	Stock items can (optionally) have color
PackageTypes	Ways that stock items can be packaged (for example, box, carton, pallet, kg, etc.
StockItemTransactions	Transactions covering all movements of all stock items (receipt, sale, write-off)
VehicleTemperatures ColdRoomTemperatures	Regularly recorded temperatures of vehicle chillers Regularly recorded temperatures of cold room chillers

Table 3.4 - Purchasing Module

Table	Description
Suppliers	Main entity table for suppliers (organizations)
SupplierCategories	Categories for suppliers (for example, novelties, toys, clothing, packaging, etc.)
SupplierTransactions	All financial transactions that are supplier-related (invoices, payments)
PurchaseOrders	Details of supplier purchase orders
PurchaseOrderLines	Detail lines from supplier purchase orders

3.4 Preparing data

To serve the given purposes and future ETL process, we need to determine the necessary data. This will avoid the excess of irrelevant data. Since our goal is to analyze customers,

products and possibly suppliers, the following data will be selected:

For Customer Data:

Data in the Sales module related to CustomerID, CustomerName, CustomerCategoryName will be retrieved. Specifically, CustomerID and CustomerName will be obtained from Sales.Customers, CustomerCategoryName will be obtained from Sales.CustomersCategories through CustomerID.

To make it easier to visualize, below is the relationship diagram in the Sales module:

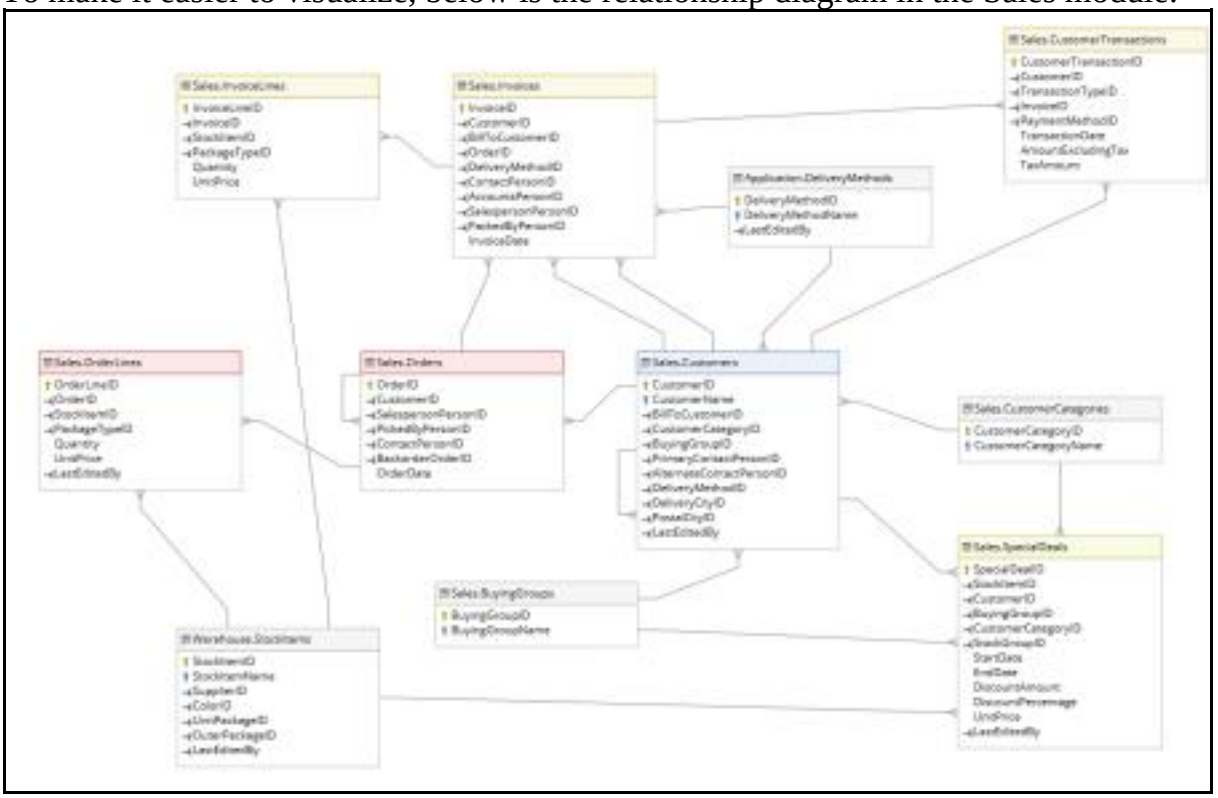

Figure 3.1 - Relational schema in Sales module

Table 3.5 - Customers data Description

Name	Source	Type Description
CustomerID	Sales.Customers	int — Numeric ID used for reference to a customer within the database (Primary key)
CustomerName	Sales.Customers	nvarchar(100) — Customer's full name (usually a trading name)
CustomersCategory	Sales.Customers Categories	nvarchar(50) — Full name of the category that customers can be assigned to

For StockItem data:

Data in the Warehouse module related to StockItemID, StockItemName, StockGroupName and UnitPrice will be retrieved. Specifically, data for the StockItemID, StockItemName and UnitPrice columns will be extracted from Warehouse.StockItem. The data of the StockGroupName column will be taken from Warehouse.StockGroup (by referencing StockItemID in the Warehouse.StockItem table to the StockGroupID of the StockItemStockGroups table to the StockGroupName of the Warehouse.StockGroups table).

To make it easier to visualize, below is the relationship diagram in the Sales module:

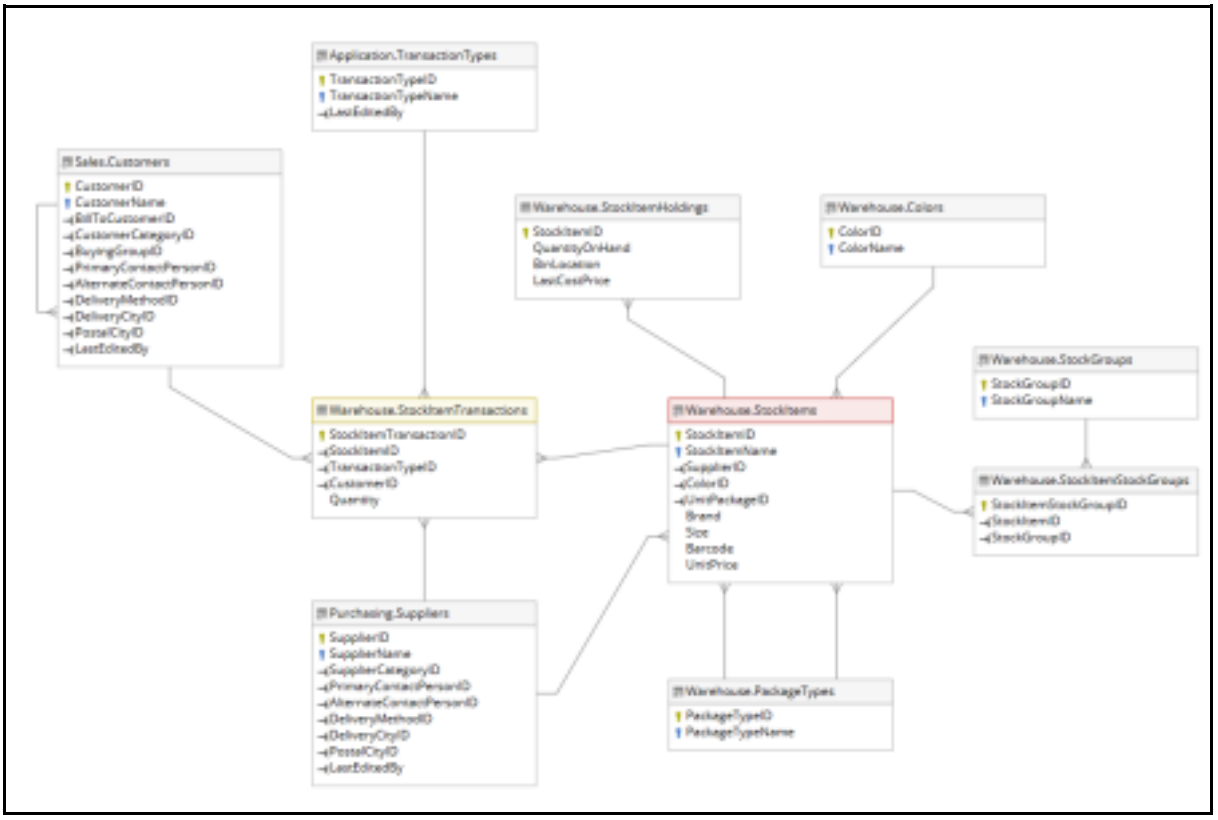

Figure 3.2 - Relational schema in Warehouse module

Table 3.6 - StockItem data Description

Name	Source Type Description

StockItemID	Warehouse.StockItem	int	Numeric ID used for reference to a stock item within the database. (Primary key)
StockItemName	Warehouse.StockItem	nvarchar(100)	Full name of a stock item.
UnitPrice	Warehouse.StockItem	decimal(18, 2)	Selling price (ex-tax) for one unit of this product

For Suppliers data:

Data in the Purchasing module related to SupplierID, SupplierName and SupplierCategoryName will be retrieved. Specifically, the SupplierID and SupplierName columns will be taken from the Purchashing.Suppliers table. The SupplierCategoryName column will be taken from the Purchashing.SuppliersCategories table via SupplierCategoryID from the

Purchashing.Suppliers table. To make it easier to visualize, below is the relationship diagram in the Purchasing module:

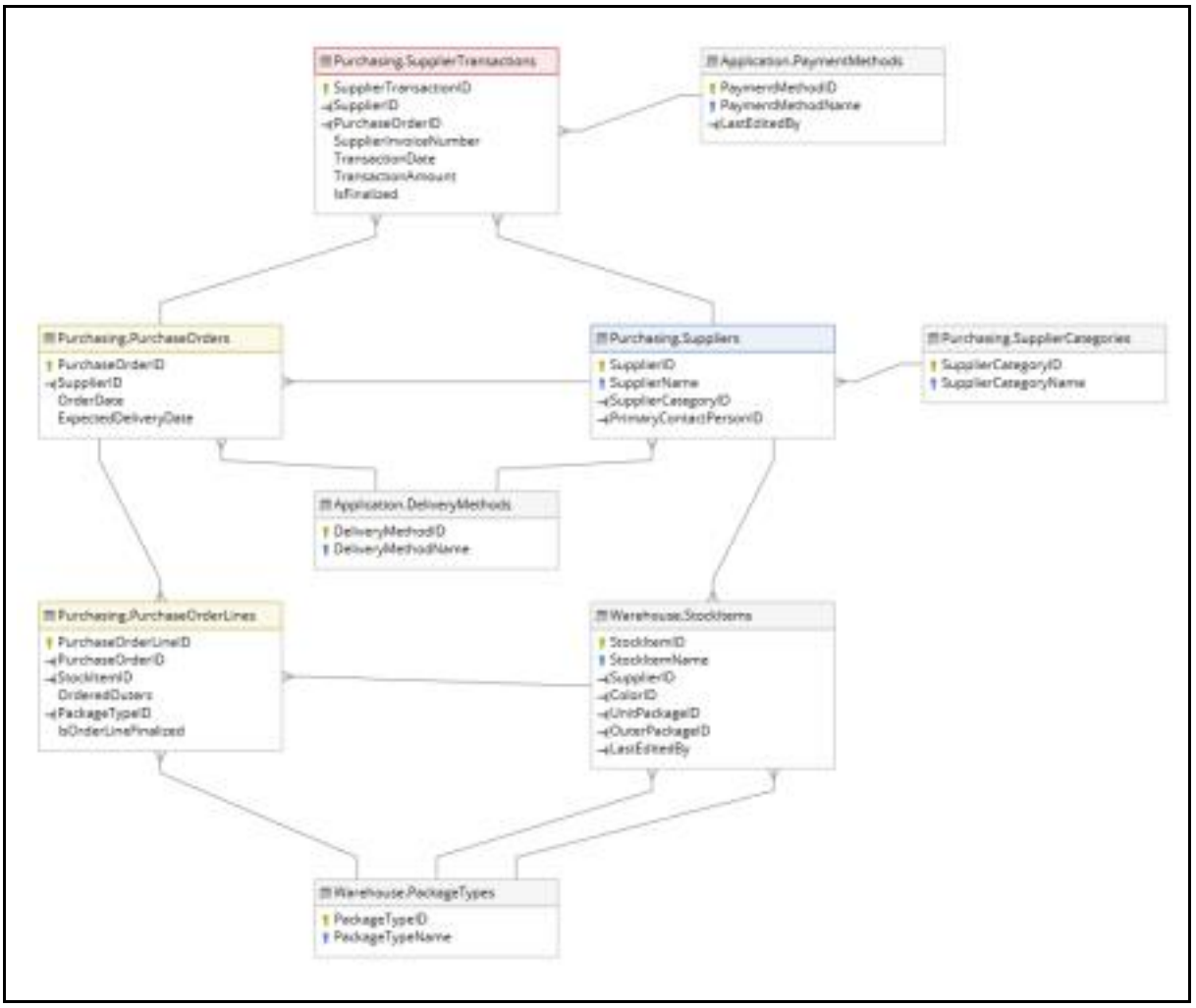

Figure 3.3 - Relational schema in Purchasing module

Table 3.7 - Suppliers data Description

Name	Source	Type Description
SupplierID	Sales.Customers	int — Numeric ID used for reference to a supplier within the database (Primary key)
SupplierName	Sales.Customers	nvarchar(100) — Supplier's full name (usually a trading name)
SupplierCategory Name	Sales.Customers Categories	nvarchar(50) — Full name of the category that suppliers can be assigned to

For City Data:

The City data we synthesize is taken from the Countries and Cities tables in the Application module.

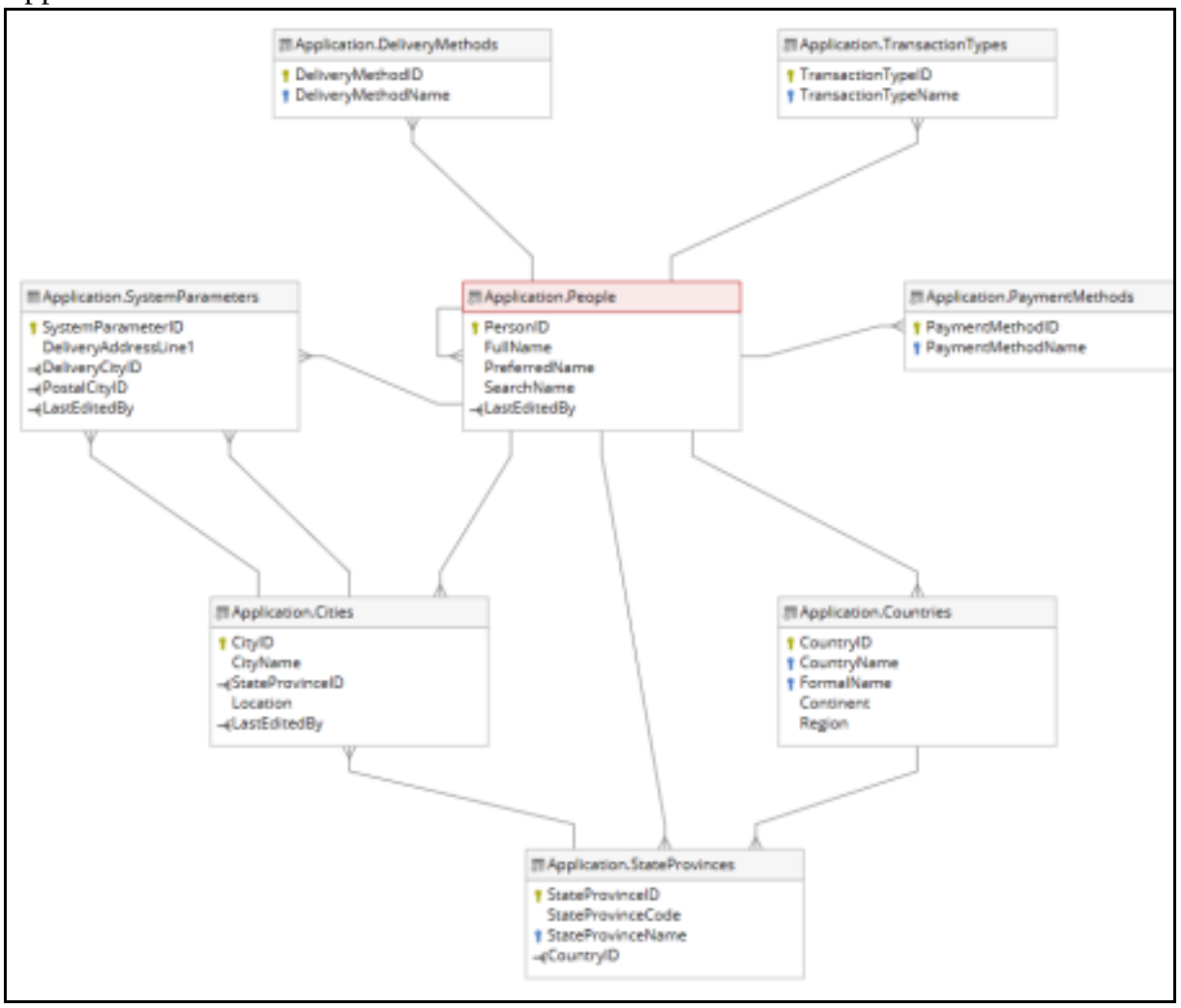

Figure 3.4 Relational schema in Application module

Table 3.8 - City data Description

Name	Source Type	Description
CityID	Application.Cities int	Numeric ID used for reference to a city within the database (Primary key)
CityName	Application.Citiesnvarchar(50)	Numeric ID used for reference to a city within the database
Location	Application.Cities geography	Geographic location of the city
CountryName	Application.Countries nvarchar	Name of the country
Continent Region	Application.Countriesnvarchar(30) Application.Countriesnvarchar(30)	Name of the continent Name of the region

For Date Data:

When designing a data warehouse, there is one dimension that is shared by all fact tables, which is the Date dimension. Measurements and events need to record the time the measurement or event occurred. This dimension is not in the WWI dataset but it will be generated by writing data generation tcode (using SQL server).

Fact Sale Table:

Finally, the Fact table. In this project, we will design the fact table in the form of "transactional fact". That is, the transaction Fact table is described by data from the Dim tables. These data do not have too high computational significance but only select important events such as number of transactions, revenue, costs, profits... available from Dim tables.

Chapter 4. Building Data Warehouse and Integrating Data

4.1 Design Data Warehouse Model

4.1.1 Busmatrix

In the star schema you provided, the business matrix could look something like this:

Business Process	Customer	Stock Item	Supplier	City	Date
Customer Segment	X	X			X
Stock Items Performance	X	X	X	X	X
Supplier Performance		X	X		X
Profitability Analysis		X		X	X

Figure 4.1- Bus Matrix

The business matrix that outlines the relationship between different business processes and various elements or dimensions that are tracked within a business. The matrix includes four business processes listed on the vertical axis: Customer Segment, Stock Items Performance, Supplier Performance, and Profitability Analysis.

On the horizontal axis, there are five dimensions that these processes might interact with or impact: Customer, Stock Item, Supplier, City, and Date. Each of these is highlighted in an orange shade, and they are set at an angle to the table below.

For each business process, there are marks (X's) in the corresponding columns that indicate which dimensions are relevant to that process.

The table is a tool typically used to visualize where different areas of a business intersect with various aspects of data collection or analysis. It helps in identifying which data is

necessary for each kind of analysis and can be used to ensure that the right data is collected and made available for the right process.

4.1.2 Star Schema of Sale

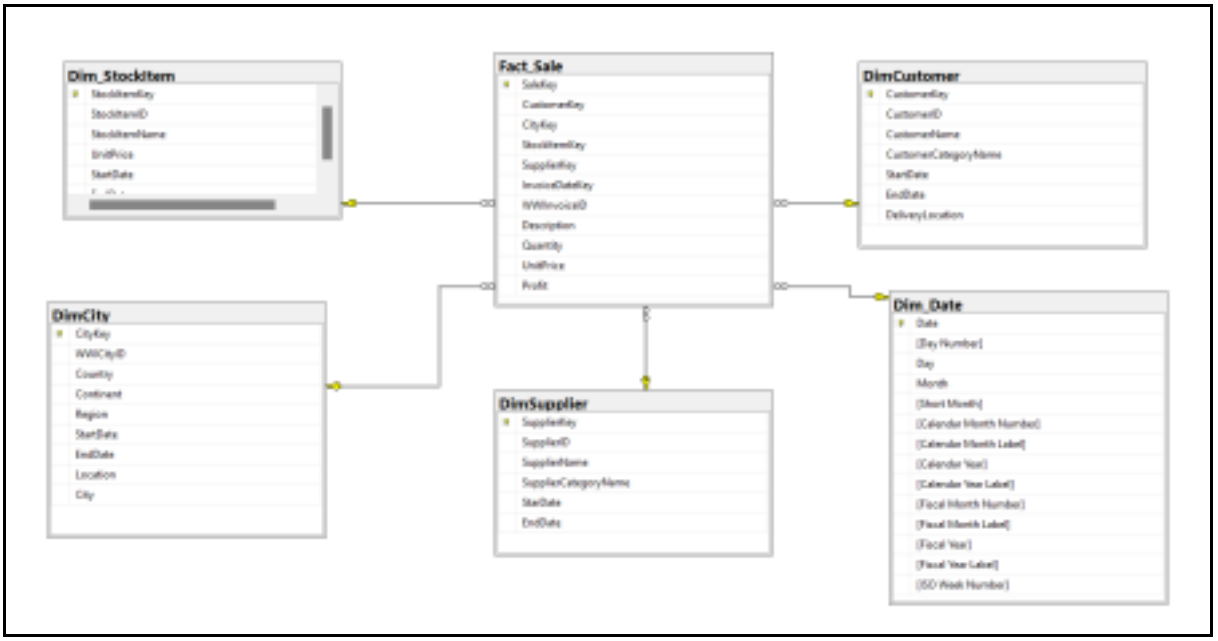

Figure 4.2 - Star Schema of Sale

The diagram you've provided appears to be a schema of a relational database, specifically designed for a business intelligence (BI) system. It's a star schema, a common approach for modeling data warehouses that facilitates complex queries and fast data retrieval. This schema shows several Dimension tables (Dim) and a Fact table (Fact), each representing different entities and metrics relevant to a business's operations.

Here's a breakdown of the diagram:
Dim_StockItem: This table stores information about stock items, likely products or goods that the business sells. Attributes like StartDate, EndDate, StockItemID, and UnitPrice suggest it tracks changes in inventory over time, including price adjustments.

Dim_Date: It contains temporal data like CalendarYear, Date, Day, FiscalYear, etc. This enables the business to perform time-based analysis and reporting, which is crucial for understanding sales patterns, seasonality, and growth trends.

Fact_Sale: This is the central table where transactional data is recorded. It includes keys from other dimension tables, such as StockItemKey, CustomerKey, and SupplierKey,

along with sales-specific metrics like Profit, Revenue, Quantity, and InvoiceDateKey. This table is used to answer questions about sales performance.

DimCustomer: Here, customer information is stored, including attributes like CustomerCategoryName, CustomerID, and DeliveryLocation. This can be used for customer segmentation, sales targeting, and tracking customer behavior.

DimSupplier: It holds details about suppliers, which is essential for managing supply chains, negotiating terms, and monitoring the performance of various suppliers.

DimCity: This likely contains geographic data about cities, including City, Country, Region, and Continent. This is valuable for regional sales analysis and market penetration studies.

For businesses, this schema can significantly aid in strategic decision-making. It allows for complex queries that combine multiple data dimensions, enabling insights into sales trends, inventory management, customer behavior, and supplier performance. The star schema is optimized for querying large data sets, making it ideal for BI applications where speed and efficiency are important.

The clear relationships between different business areas facilitate deep analytics, such as identifying which products are most profitable, which suppliers are most reliable, and which customer segments are driving the most revenue. Overall, such a database schema is a cornerstone for any data-driven business strategy, providing a robust foundation for reporting, analysis, forecasting, and ultimately driving business growth.

4.2 ETL Process

4.2.1 Design ETL process

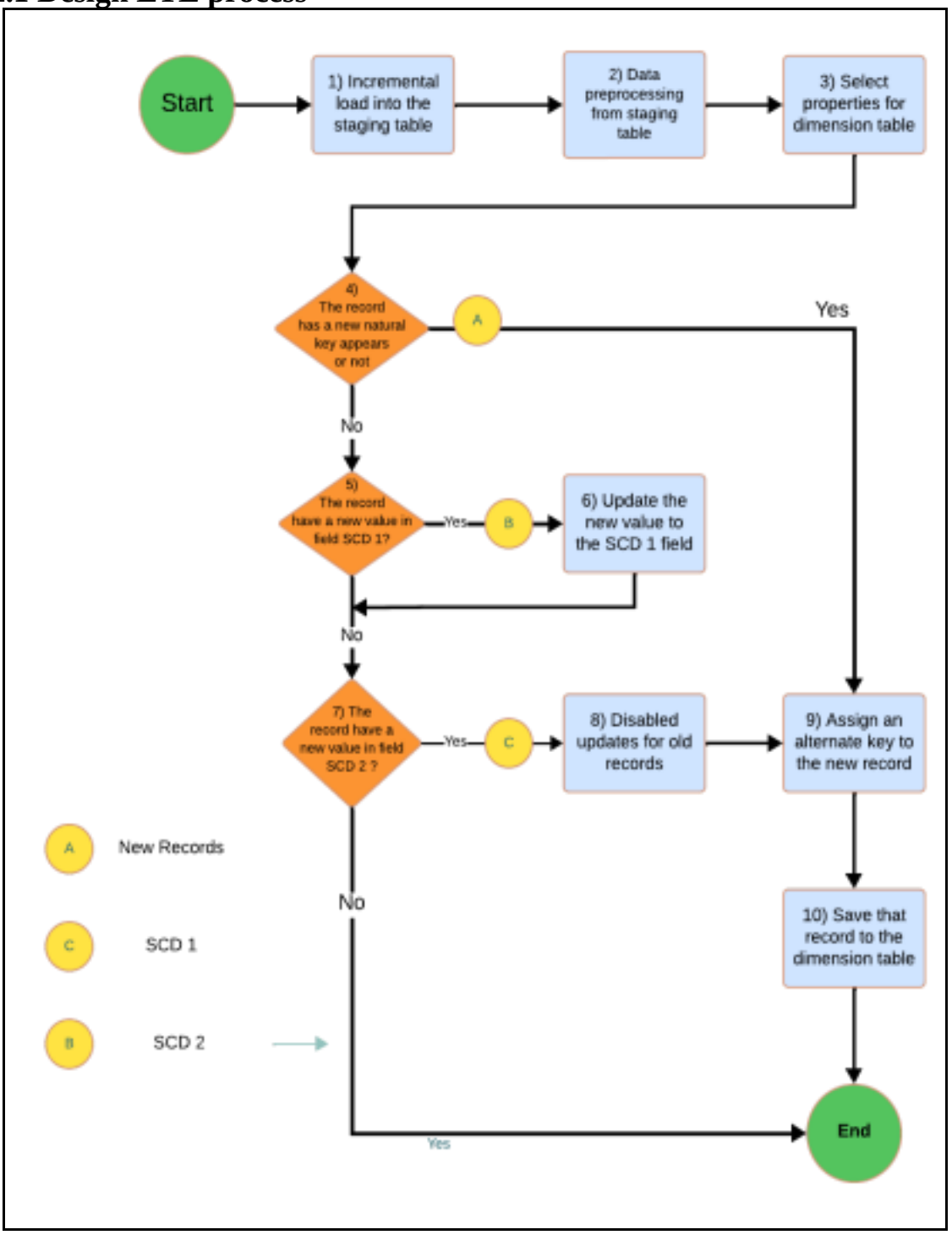

Figure 4.3 - Data dimension processing workflow

Describe the ETL Process:

(1) Step 1 of the dimension processing workflow involves synchronizing data from the business system into staging tables. This synchronization can be done in either incremental form (only retrieving the latest records and processing them

synchronously using ETL) or full synchronization (fetching all records, including those obtained in the previous synchronization but not yet updated). Incremental synchronization is feasible only for tables with primary keys and storing the last update timestamp for each record, making it challenging and posing risks of inconsistency. Full synchronization retrieves all records in the data table, offering a simple processing solution but putting a burden on the business system. ETL developers need to choose a reasonable approach that balances development and operational processes for the data warehouse and the system's capacity.

(2) Preprocess data from the business system based on dimension rules, such as standardizing units of measure, handling duplicate records, and flattening attributes (in cases where the source table follows the EAV model).

(3) Re-create all objects in the dimension table in their latest state.

(4) Check the temporary table from step (3) to see if there are any new dimension objects by examining the natural key. If new objects are found, assign replacement keys for those objects (step 9) and save the records to the dimension table (step 10).

(5) For existing dimension objects, check whether the SCD 1 data column has new values. If there are new values, update that column in the dimension table (step 6).

(6) In the processing flow for existing dimension objects, check whether the SCD 2 data column has new values. If there are no new values, stop the ETL process. If there are new values, update the status of existing records to "Inactive" ('I') (step 8), assign replacement keys for the new record (step 9), and then save that record to the dimension table (step 10).

This process applies to both SCD Type 1 and Type 2, with adjustments possible for SCD Type 3 and Type 4. But in this project, my group only used SCD types 1 and 2 for all the records.

4.2.2 ETL process with SSIS

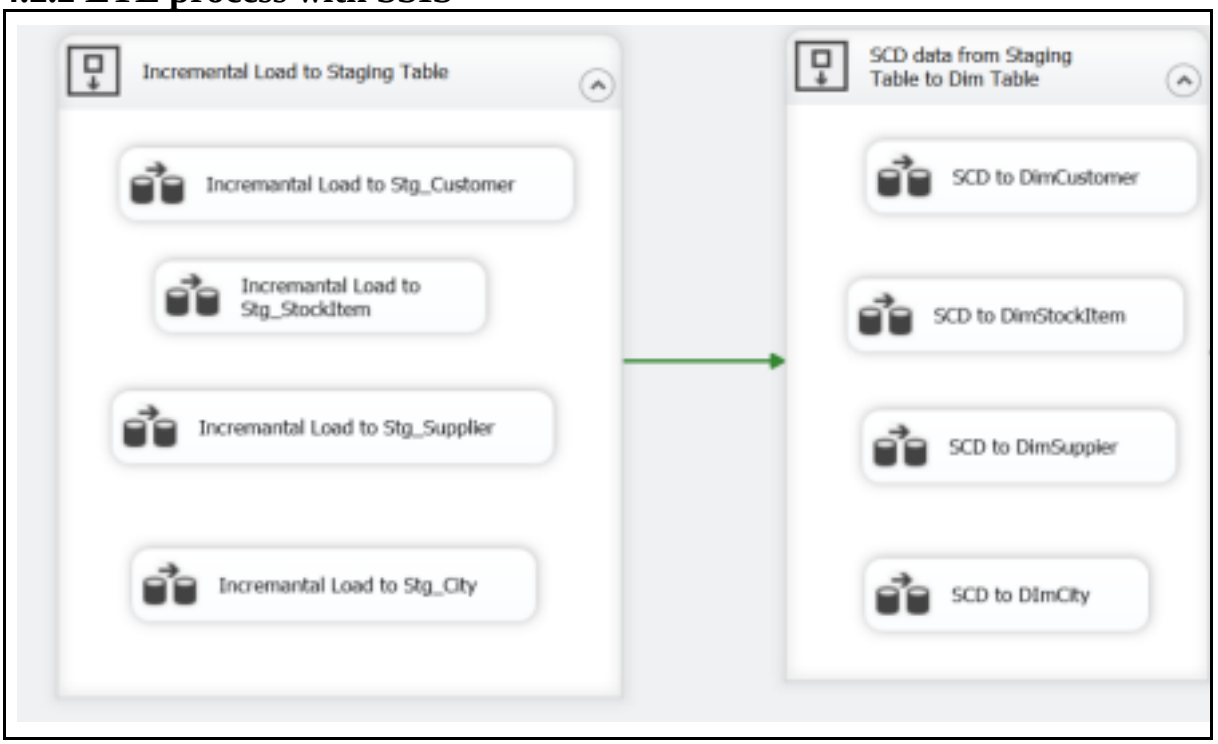

Figure 4.4 - Control Flow for ETL

In this Control Flow, there are two main parts of the ETL process used through SQL Server Integration Services (SSIS). Firstly, the 'Incremental Load to Staging Table' comprises four processes for incrementally loading data into temporary tables for stg_Customer, stg_StockItem, stg_Supplier, and stg_City. This section ensures that only new or updated data since the last load is brought in, optimizing system resources and operational efficiency. Subsequently, data from these temporary tables is transferred to dimension tables through Slowly Changing Dimension (SCD) steps. This process utilizes SCD to accurately update information in dimension tables, preserving data history and reflecting changes in a consistent manner. This provides a reliable database for business analysis and decision-making based on data, while keeping the data warehouse constantly updated and intact.

Create Dim Customer

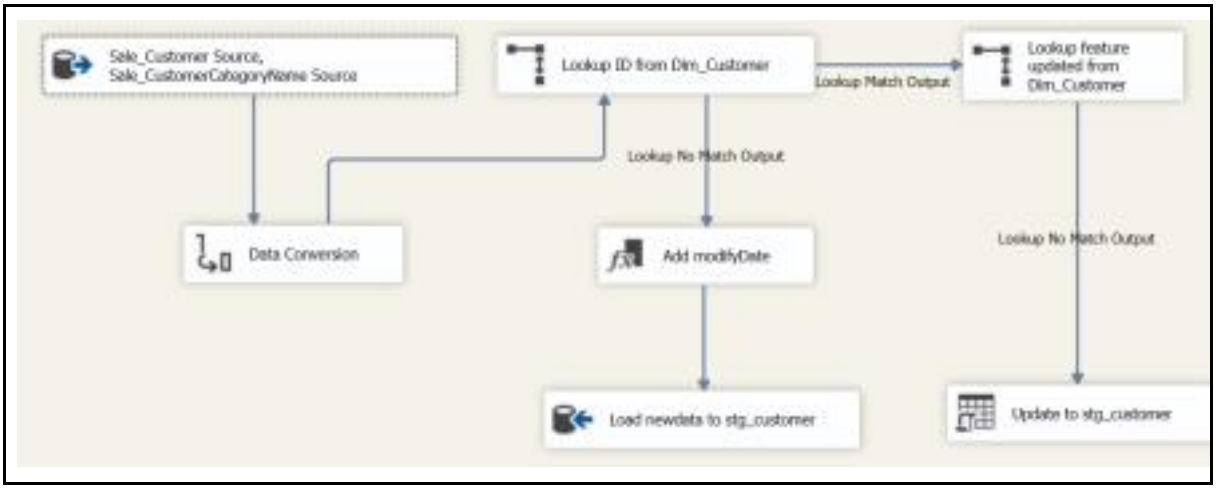

Figure 4.5 - Data flow of Incrementa load to Customer Staging Table

In the OLDB Source, write a SQL Command to join the Sale.Customer table and the Sale.CustomerCategory table, then select the necessary columns to load into the Customer Staging table. After a few Data Conversion steps, the data will be passed to the Lookup Transform for comparing the data to determine which records will be incrementally loaded into the Customer Staging table.

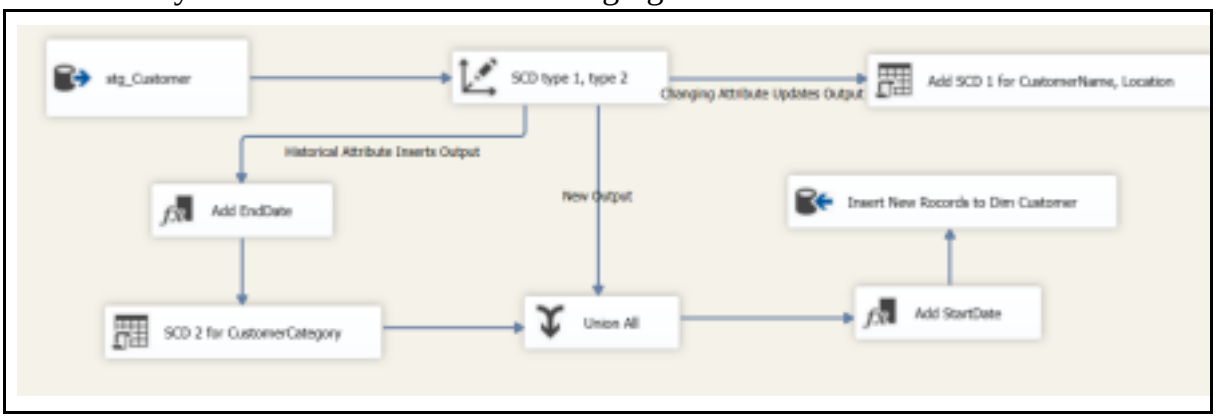

Figure 4.6 - Apply SCD Type 1 and 2 for DimCustomer

CustomerCategory needs to be tracked for historical changes over time. Changes in CustomerCategory can reflect shifts in customer interaction levels or strategies. SCD Type 2 provides a clearer understanding of how customers react and evolve over time.

We apply SCD Type 1 to CustomerName and Location since we only need to focus on their current values, and the data type of Location is too complex to store its history with meaningful insights. SCD Type 1 ensures that we capture and maintain the current state of these attributes without tracking their historical changes.

Create Dim StockItem

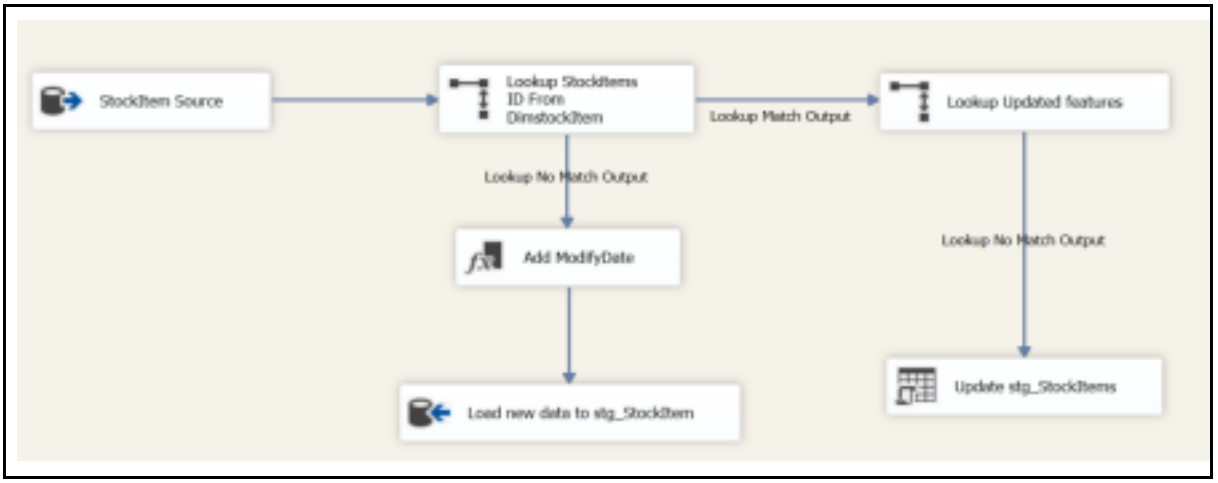

Figure 4.7 -Data flow of Incrementa load to Staging StockItems Table

For the StockItems Table, the source data comes solely from the Warehouse.StockItems table. Using the Lookup Transform, the data will be compared based on StockItems ID to identify new IDs. If new IDs are found, they will be inserted into the Staging StockItems table. If no new IDs are identified, the data will be updated if there are any changes.

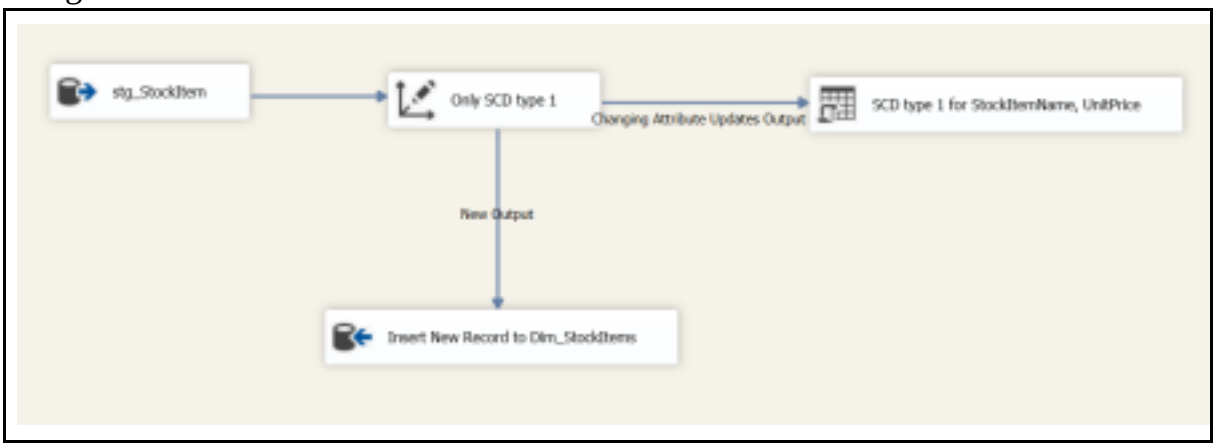

Figure 4.8 - Apply SCD type 1 DimStockItem

We use SCD Type 1 for all columns in the Dim StockItems. SCD Type 1 preserves the current values of StockItemName and UnitPrice. If there are any changes, their current values will be overwritten, keeping the dimension table current and straightforward. Since there is no need to track the historical changes of StockItemName and UnitPrice, SCD Type 1 is the appropriate choice, helping to keep the data structure simple and focused on current information.

Create Dim Supplier

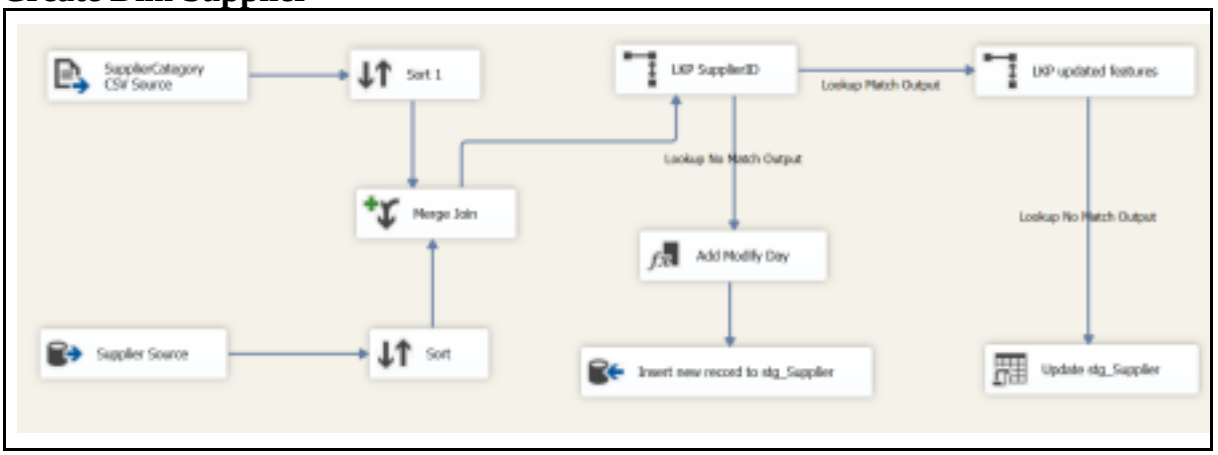

Figure 4.9 - Data flow of Incrementa load to Staging StockItems Table

In this Data Flow, the source data comes from two tables: SupplierCategory from a CSV file and Supplier Source from an OLDB source. Subsequently, these two tables are merged using the SupplierCategoryID, and the combined data is passed through a Lookup Transform for standard processing, similar to other Data Flows. Finally, an Incremental Load is performed into the Staging Supplier table.

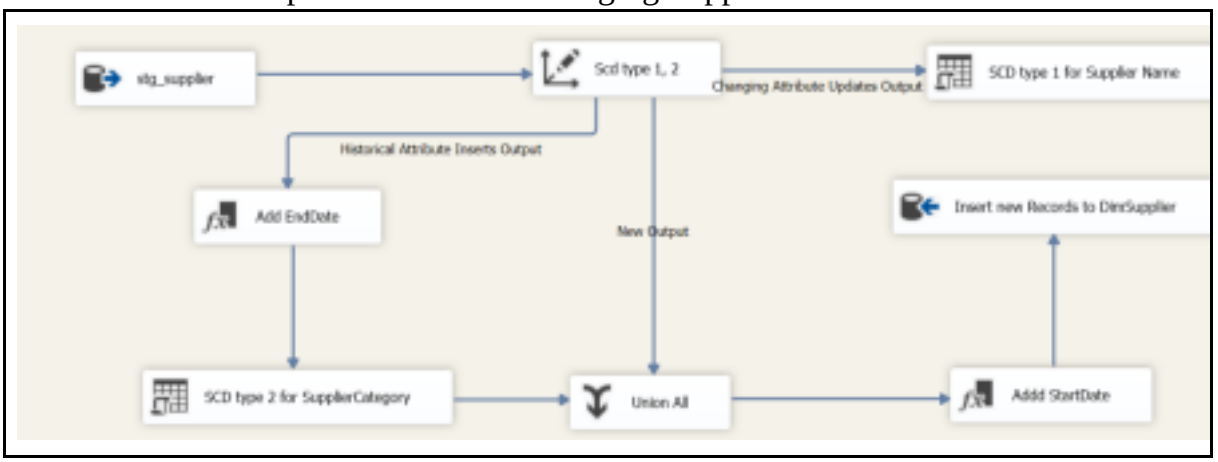

Figure 4.10 - Apply SCD Type 1 and 2 for DimSupplier

SupplierCategory needs to be tracked for historical changes over time. Changes in SupplierCategory can reflect shifts in the interaction level or strategy of product supply sources. SCD Type 2 helps provide a deeper understanding of Suppliers and optimize strategies related to input products.

We apply SCD Type 1 to SupplierName because maintaining the history of this attribute doesn't provide significant insights for the data analysis process.

Create Dim City

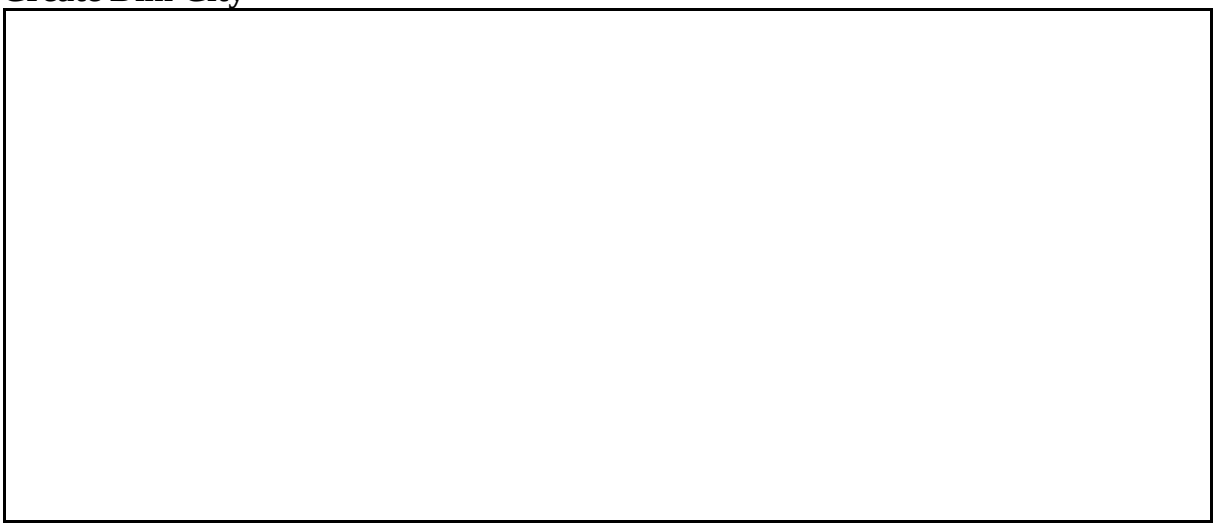

Figure 4.11 - Data flow of Incrementa load to Staging City Table

In this Data Flow, the most challenging part is likely the join between the City Source table and the Country Source table using the Location column of the City table and the Border column of the Country table. Handling data from these two columns in the form of Location can be challenging compared to the more straightforward processing in other Data Flows mentioned earlier.

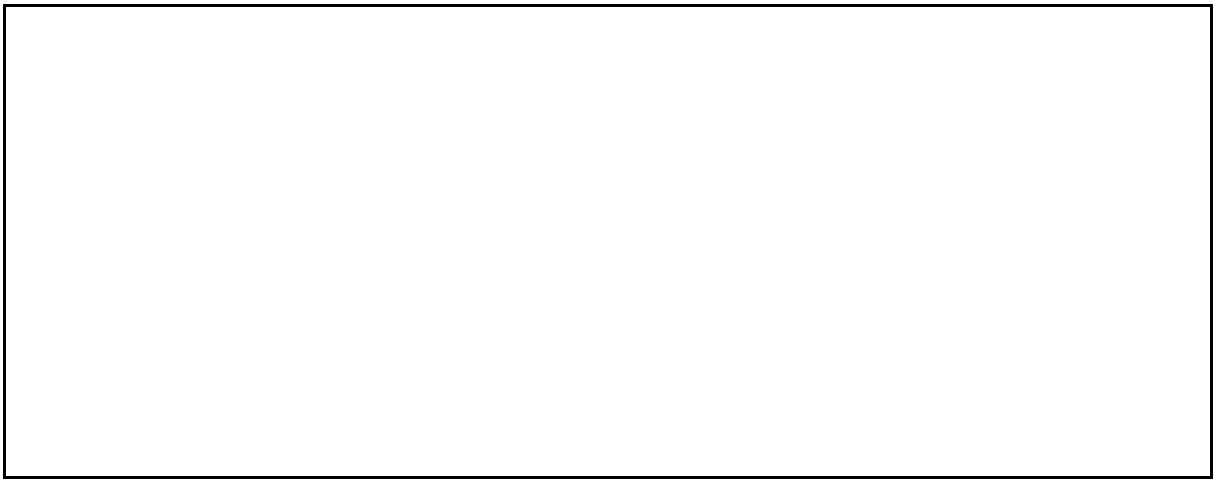

Figure 4.12 - Apply SCD Type 1 and 2 for DimCity

We apply SCD Type 2 to CityName because we want to track when customers leave a city or possibly when customers frequently move between two cities.

As for columns like Country, Continent, and Location, we should apply SCD Type 1. This is because, with SCD Type 2 applied to CityName, we already have sufficient information about the customer's address. Additionally, dealing with the Geography datatype of the Location column makes it challenging to implement SCD

Type 2.

Create Dim Date

When designing a data warehouse, there is a shared dimension for all fact tables, namely the Date dimension. Measurements and events all require recording the time they occurred. This dimension is not sourced from the operational system but is typically generated by writing code (using SQL, Excel, or any programming language). The Date dimension table is created once at the beginning of the project and remains unchanged throughout the entire lifecycle of building, operating, and upgrading the data warehouse. Changes, if any, occur periodically once a year due to certain holidays in Vietnam that follow the lunar calendar and do not have fixed Gregorian dates.

Unlike other dimension tables that use surrogate keys, the Date dimension table is often designed with a meaningful surrogate key, encoded from the date of the record in the format YYYYMMDD. For example, a record for May 23, 2019, might have a DIM_ID value of 20190523. Below is the SQL script to create the DimDate table:

```
DECLARE @StartDate date = '20110101';
DECLARE @Year int = 4;
DECLARE @CutoffDate date = DATEADD(DAY, -1, DATEADD(YEAR,
@Year, @StartDate));

;WITH seq(n) AS
(
  SELECT 0 UNION ALL SELECT n + 1 FROM seq
  WHERE n < DATEDIFF(DAY, @StartDate, @CutoffDate)
),
d(d) AS
(
  SELECT DATEADD(DAY, n, @StartDate) FROM seq
),
src AS
(
  SELECT
    DateKey = CONVERT(date, d),
    TheDay = DATEPART(DAY, d),
    TheDayName = DATENAME(WEEKDAY, d),
    TheWeek = DATEPART(WEEK, d),
    TheISOWeek = DATEPART(ISO_WEEK, d),
    TheDayOfWeek = DATEPART(WEEKDAY, d),
    TheMonth = DATEPART(MONTH, d),
    TheMonthName = DATENAME(MONTH, d),
    TheQuarter = DATEPART(Quarter, d),
```

```
    TheYear = DATEPART(YEAR, d),
```



```
    TheFirstOfMonth = DATEFROMPARTS(YEAR(d), MONTH(d), 1),
    TheLastOfYear = DATEFROMPARTS(YEAR(d), 12, 31),
    TheDayOfYear = DATEPART(DAYOFYEAR, d)
  FROM d
)

SELECT *
INTO DimDate
FROM src
ORDER BY DateKey
OPTION (MAXRECURSION 0)
```

4.2.3 Cleaning Staging Table

Deleting data from the staging table periodically has several benefits in managing system data and the ETL (Extract, Transform, Load) process. The staging table frequently receives new data from various sources. As the volume of data in the staging table increases, queries and operations on this table may become slower. Deleting old data helps reduce the size of the table and improves query performance.

Deleting old data also helps reduce the storage capacity of the database. This can be crucial for systems with limited storage space. Reducing the amount of retained data in the staging table can lower data storage costs, especially when using storage services based on capacity.

Deleting old data from your Staging Table requires a clear definition of the date when the data was loaded into the staging table or modified. Below is an example code snippet for creating a trigger to delete data from the stg_Customer table for data with a ModifyDate greater than 30 days:

```
CREATE TRIGGER tr_Staging_Customer_Cleanup
ON stg_Customer
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    -- Delete records older than 30 days
    DELETE FROM stg_Customer
    WHERE ModifyDate < DATEADD(DAY, -30, GETDATE());
END;
```

4.2.4 Fact Sale Table

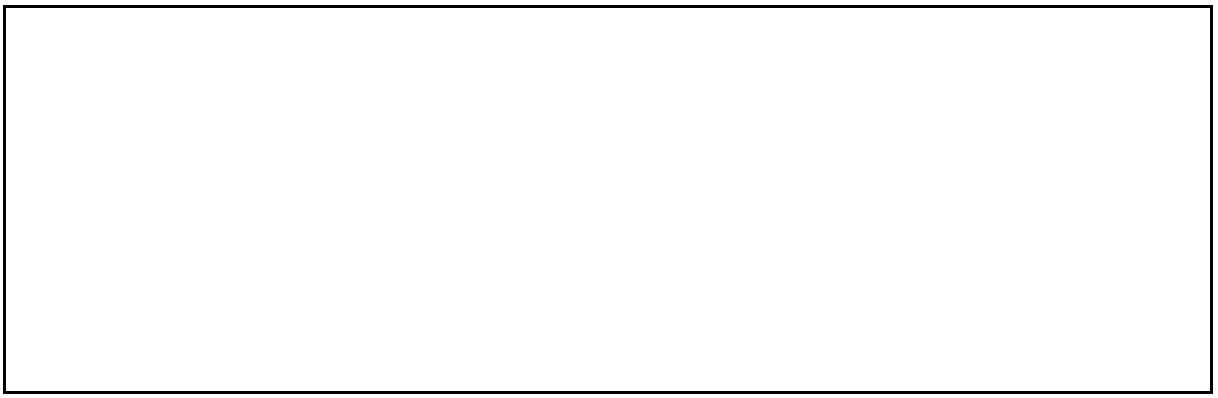

Figure 4.13 - Data Flow Loading Dim Table into Fact Table

The data is retrieved from the InvoiceLine Source. Using a lookup transformation (LKP DimCustomer), SSIS compares and searches for foreign keys from the data source based on customers (DimCustomer). The purpose is to determine a unique key (e.g., CustomerID) for each invoice line corresponding to a specific customer in the customer dimension table. Lookups for other Dimensions: A similar process is performed with other dimension tables such as DimSupplier, DimStockItem, and DIM_Date. Each lookup identifies a unique key for each dimension table based on data from the source.

Once the keys from the dimension tables are obtained, the data rows are joined back together through the outputs of the lookups (Lookup Match Output). This ensures that each invoice line will have complete corresponding keys from each related dimension table. Meanwhile, another invoice source (LKP Invoice Source) can be used to retrieve additional information needed to create an event table, such as InvoiceID or details related to the total amount, taxes, etc.

Finally, after all necessary keys and data have been identified and joined back, the data is inserted into the Fact_Sale table through an Insert task. This event table will store aggregated information about sales, including keys from dimension tables and metrics such as revenue, profit, quantity sold, etc.

Chapter 5. Analysis and Recommendations for Businesses

5.1 Overall the Analysis Results with Power BI:

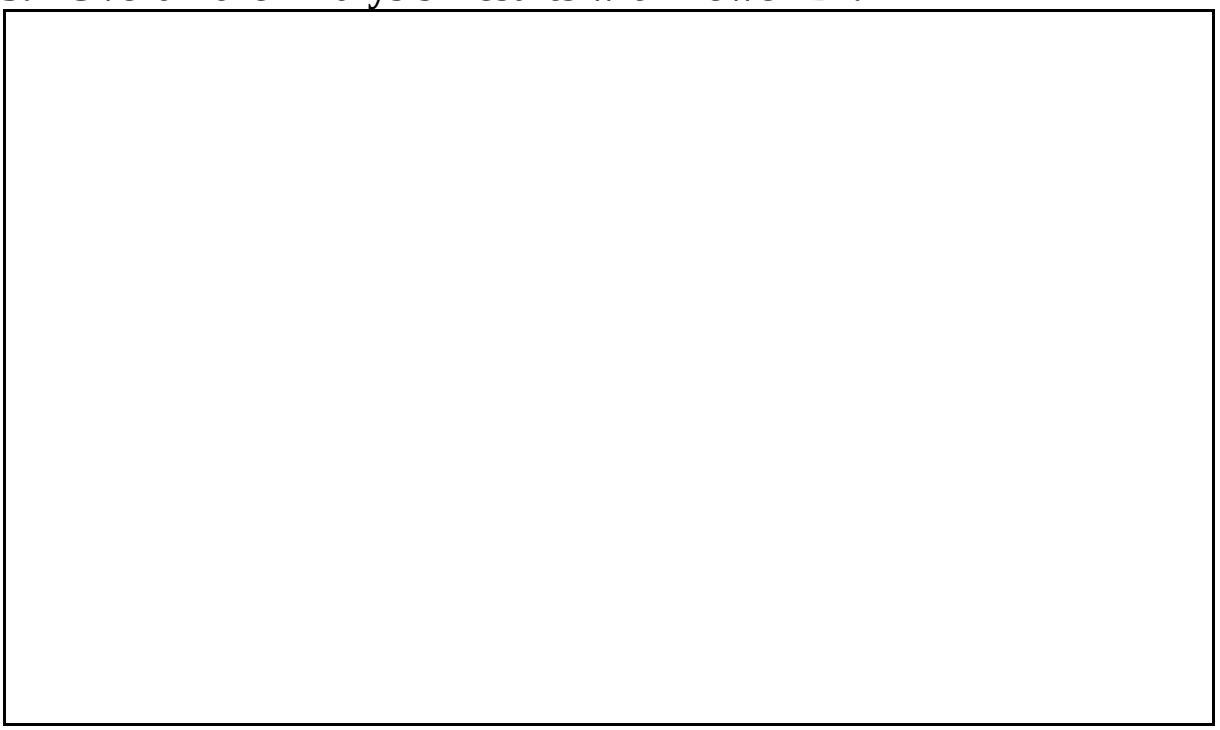

Figure 5.1 Overall the Analysis Results with Power BI

The results table above shows an overview of the data after the data warehouse construction process. First, the total revenue that WWI had during the data collection period was 172.26M. A positive number within 4 years. The profit rate reaches 49.8%. With an average value of each order of 754.66 for store customers, this is not too high a threshold, WWI needs to improve this. In addition, aspects such as top selling and unsold products and important customers are also shown on the report.

Click on the following link to learn more about Power BI: Microsoft Power BI

5.2 Business question and Recommendation

In the final stage of this project, to have a broader view of the WWI company's business situation, we will perform multidimensional data queries with the MDX language. Of course, then there is data visualization to be more clear in making business proposals for businesses.
We will query the data based on the initial goals set:

(1) What are the 6 months with the highest total revenue over the years?

Code MDX:

```
select
      [Measures].[Profit] on columns,
     TopCount([Dim Date].[Month].[Month].Members, 6,
 [Measures].[Profit]) on rows
from [WWI DW]
```

Result:

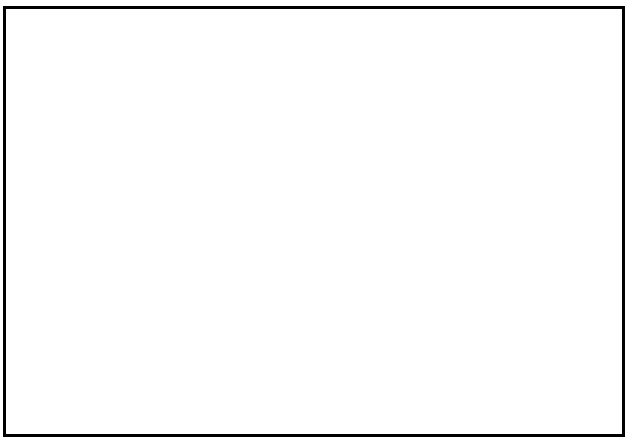

Figure 5.2 Result of question 1

Recommendations: The results show that the most profitable months of the year are January, February, March, April, May and July. Based on the trend of high profits (first 3 months) in the first months of the year and peak periods in the season (April, May and July), we should focus on strengthening marketing and advertising strategies during these months to take advantage of increased shopping demand from customers. Besides, we should also consider whether to apply this business model to other times of the year.

(2) Where are "best-selling products" mainly located?

Code MDX:

```
SELECT
TOPCOUNT(
NONEMPTY(
 [Dim Stock Item].[Stock Item Name].[Stock Item Name].MEMBERS,
[Measures].[Quantity]), 5, [Measures].[Quantity]) ON rows,
TOPCOUNT(
NONEMPTY(
 [Dim City].[City].[City].MEMBERS, [Measures].[Quantity]), 6,
 [Measures].[Quantity]) ON columns
FROM [WWI DW]
WHERE ([Measures].[Quantity])
```

Result:

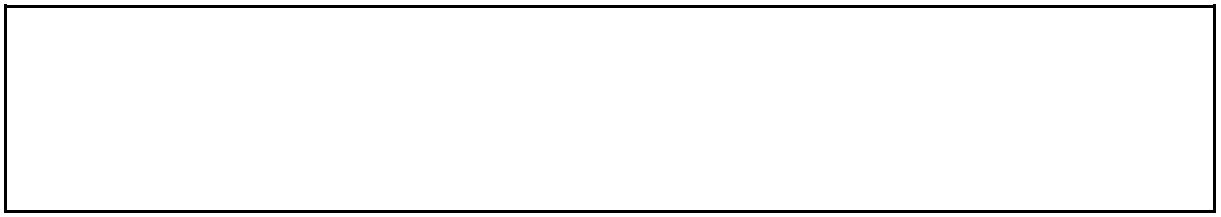

Figure 5.3 Result of question 2

Recommendations: Based on the above results, we recommend focusing on optimizing inventory and advertising this product in high concentration areas. At the same time, you may want to consider expanding your distribution footprint or enhancing your marketing strategy in those areas to maximize sales and take advantage of purchasing demand in these key locations.

(3) Where are the "most profitable products" mainly concentrated?
Code MDX:

```
SELECT
TOPCOUNT(
NONEMPTY(
 [Dim Stock Item].[Stock Item Name].[Stock Item Name].MEMBERS,
[Measures].[Quantity]), 5, [Measures].[Profit]) ON rows,
TOPCOUNT(
NONEMPTY(
 [Dim City].[City].[City].MEMBERS, [Measures].[Profit]), 6,
 [Measures].[Profit]) ON columns
FROM [WWI DW]
WHERE ([Measures].[Profit])
```

Result:

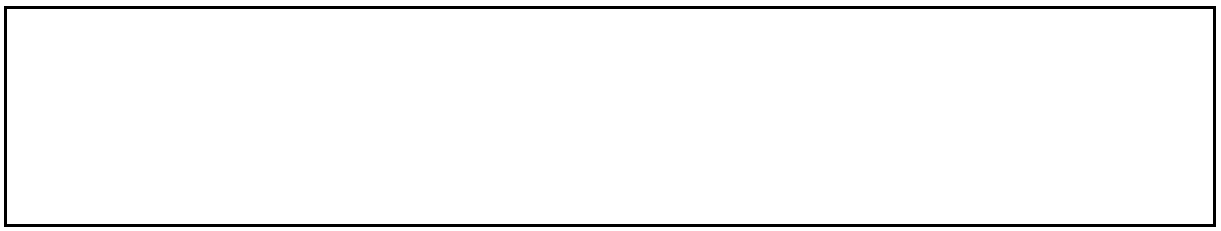

Figure 5.4 Result of question 3

Recommendations: Although the above products do not have a large sales volume, they bring a large profit to the company. Therefore we should strengthen marketing and advertising strategies for these products in these locations. At the same time, you can

consider expanding or optimizing distribution channels to maximize revenue from these strategic products.

45
46

(4) Statistics on purchasing frequency of customer groups each year
Code MDX:

```
with member [Dim Date].[Fiscal Year].[Total] as
aggregate(
{
[Dim Date].[Fiscal Year].&[2013],
[Dim Date].[Fiscal Year].&[2014],
[Dim Date].[Fiscal Year].&[2015],
[Dim Date].[Fiscal Year].&[2016]
}
)
select
[Dim Customer].[Customer Category Name].[Customer Category
Name].members on columns,
{
[Dim Date].[Fiscal Year].&[2013],
[Dim Date].[Fiscal Year].&[2014],
[Dim Date].[Fiscal Year].&[2015],
[Dim Date].[Fiscal Year].&[2016],
[Dim Date].[Fiscal Year].[Total]
} on rows
from [WWI DW]
where [Measures].[Fact Sale Count]
```

Result:

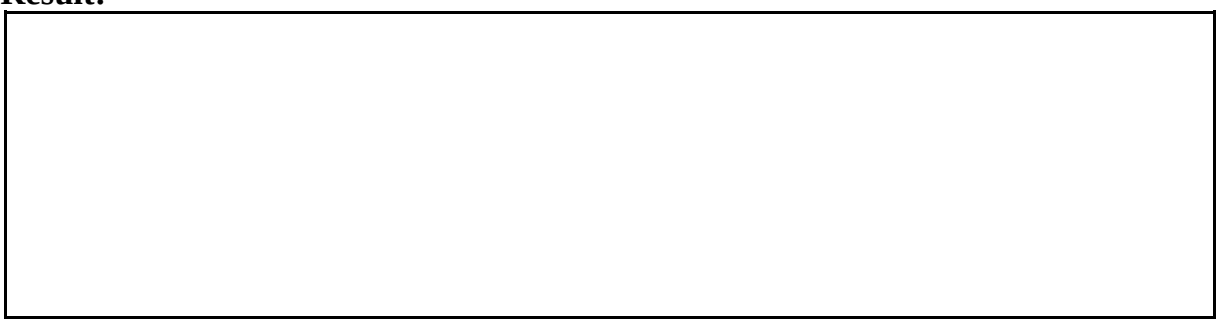

Figure 5.5 Result of question 4

Recommendations: Novelty Shop is our largest customer and Corporate is the customer with the least frequency of transactions with WWI. Our proposal is to consider the needs

of consumers and the business models of the customers we are supplying to in order to best manage inventory and devise a reasonable relationship between customers with whom WWI is dealing.

(5) Which suppliers does the best-selling products belong to?

Code MDX:

```
WITH
SET Top5Suppliers AS
TOPCOUNT(
NONEMPTY([Dim Supplier].[Supplier Name].[Supplier
Name].MEMBERS, [Measures].[Quantity]),
5,
 [Measures].[Quantity]
)
SELECT
FILTER(
TOPCOUNT(
NONEMPTY(
 [Dim Stock Item].[Stock Item Name].[Stock Item Name].MEMBERS,
[Measures].[Quantity]), 5, [Measures].[Quantity]), NOT
IsEmpty([Measures].[Quantity])
) ON ROWS,
{[Top5Suppliers]} ON COLUMNS
FROM [WWI DW]
```

Result:

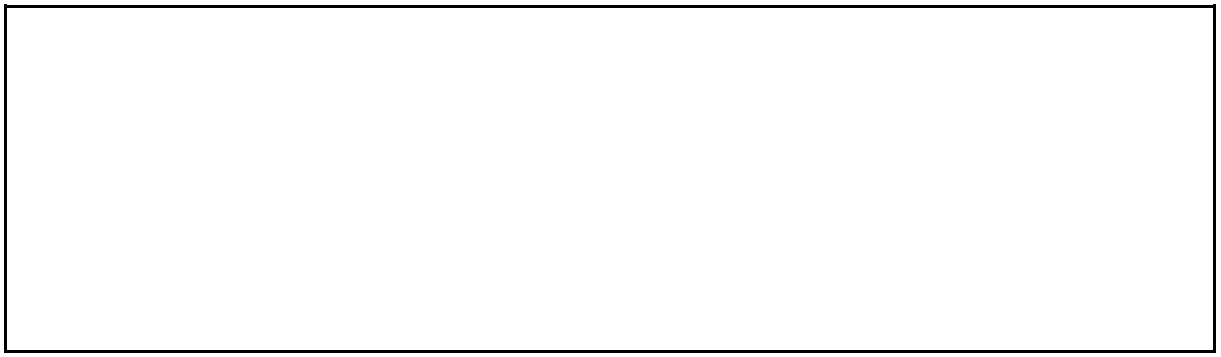

Figure 5.6 Result of question 5

Recommendations: Surprisingly, the best-selling products all belong to the supplier "Litware, lnc". This proves that products from this supplier have very good quality or

affordable prices, so they are chosen by most consumers. WWI should have a good relationship with “Litware, lnc” and review relationships with suppliers of products that are not really doing business.

References

[1] Data warehouse là gì? Kiến thức cơ bản về kho dữ liệu
[2] Kho dữ liệu – “Chiếc dạ dày” của hệ thống BI
[3] Các bước xây dựng Data Warehouse (xây dựng kho dữ liệu)
[4] 11 bước để Triển khai kho dữ liệu (DWH)
[5] Phương pháp tiếp cận, Kế hoạch, và chi phí xây dựng kho dữ liệu (Data Warehouse) [6] Mô hình Data Warehouse: Sơ đồ ngôi sao
[7] [Nhập môn Data Warehouse] Mô hình dữ liệu đa chiều
[8] Data Warehouse – Có phức tạp như bạn nghĩ?
[9] Kiến trúc DWH: Data Warehouse Architecture
[10] “ĐỊNH DANH” DỄ HIỂU VỀ DATA LAKE VÀ DATA WAREHOUSE CHO NHÀ QUẢN TRỊ DOANH NGHIỆP