



# ECOMMERCE ANALYTICS DASHBOARD

## MET CS 779 Term Project Report

### Abstract

This project delivers a streamlined E-commerce Analytics Dashboard built on a large retail dataset of 10,000+ records. It features a fully normalized OLTP schema in MySQL with 15 relational tables, and a Python-based ETL pipeline to clean and load data. Stored procedures were implemented to transform and transfer data from the OLTP schema into a dimensional star schema for OLAP analysis. Performance tuning techniques, including indexing and query optimization, were applied to improve OLTP efficiency. The OLAP schema incorporates Slowly Changing Dimensions (SCD Types two and three) to track evolving product and store attributes. SQL views provide insights into monthly sales, promotions, and product trends, while interactive dashboards in Tableau support data-driven business decisions.

Kunal Vishwa Sivakumar  
kvishwa@bu.edu

# MET CS 779 Term Project

## Table of Contents

1. Introduction .....	2
2. Project Overview .....	2
1) Dataset Overview .....	2
2) Normalization .....	2
3) Star Model:.....	3
i) Dimensional Tables .....	4
ii) Fact Table.....	5
4) Stored Procedure:.....	5
5) Performance Optimization .....	5
6) SCD Type 2 and Type 3 : With Examples.....	6
7) Views And Tableau.....	7
3. Conclusion .....	8

## 1. Introduction

In today's data-driven retail landscape, organizations rely heavily on robust backend systems and analytics platforms to uncover actionable insights from large volumes of transactional data. This project focuses on designing and implementing an end-to-end E-commerce Analytics system using MySQL as the backend relational database, supported by Python for ETL and Tableau for visualization. The core objective was to transform raw sales and operations data from a retail dataset containing over 10,000 records into a structured, normalized OLTP schema and then build a star schema for efficient OLAP analysis.

The project involved multiple key phases: data cleaning and ingestion using Python, designing normalized relational tables that adhere to BCNF/3NF standards, developing stored procedures to automate data transformation, implementing Slowly Changing Dimensions (SCD Types 2 and 3) to track historical and evolving attributes, and finally, generating interactive dashboards to support business decision-making. This practical implementation bridges data engineering and business intelligence, simulating real-world workflows found in modern enterprise data environments.

## 2. Project Overview

### 1) Dataset Overview

The dataset used in this project is based on Walmart's retail transactions, consisting of over 10,000 records simulating real-world store-level operations. Each row captures a single transaction with attributes spanning customer demographics, product details, inventory levels, promotions, payment methods, weather conditions, and store metadata. Provided as a flat CSV file, the dataset required preprocessing to extract and separate entities for normalization. Key fields such as `transaction_id`, `product_id`, `store_id`, `customer_id`, `quantity_sold`, `unit_price`, `promotion_type`, and `stockout_indicator` enabled the construction of a relational schema and supported analytics on sales performance, demand forecasting, and promotion impact.

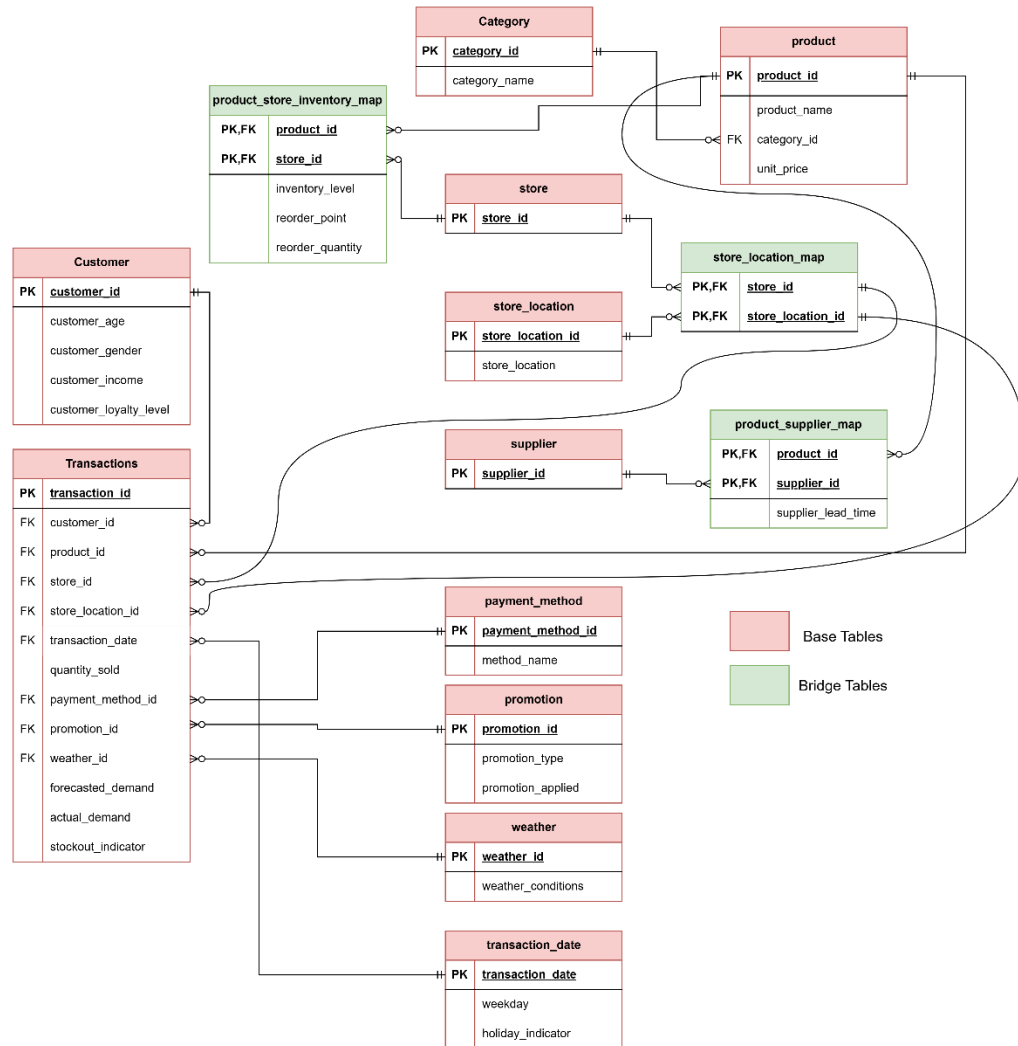
### 2) Normalization

The original Walmart dataset, provided in a flat file format, contained repeated and nested information across various attributes such as customer details, product metadata, store information, and transactional data. To improve data quality and eliminate redundancy, the dataset was decomposed into a fully normalized OLTP schema comprising 15 tables. The schema design followed principles up to the Boyce-Codd Normal Form (BCNF), ensuring that each table held atomic data with minimal duplication and clearly defined relationships.

Entities such as customer, product, store, supplier, and promotion were separated based on functional dependencies, while many-to-many relationships were resolved using bridge tables like `product_supplier_map` and `store_location_map`. All tables were

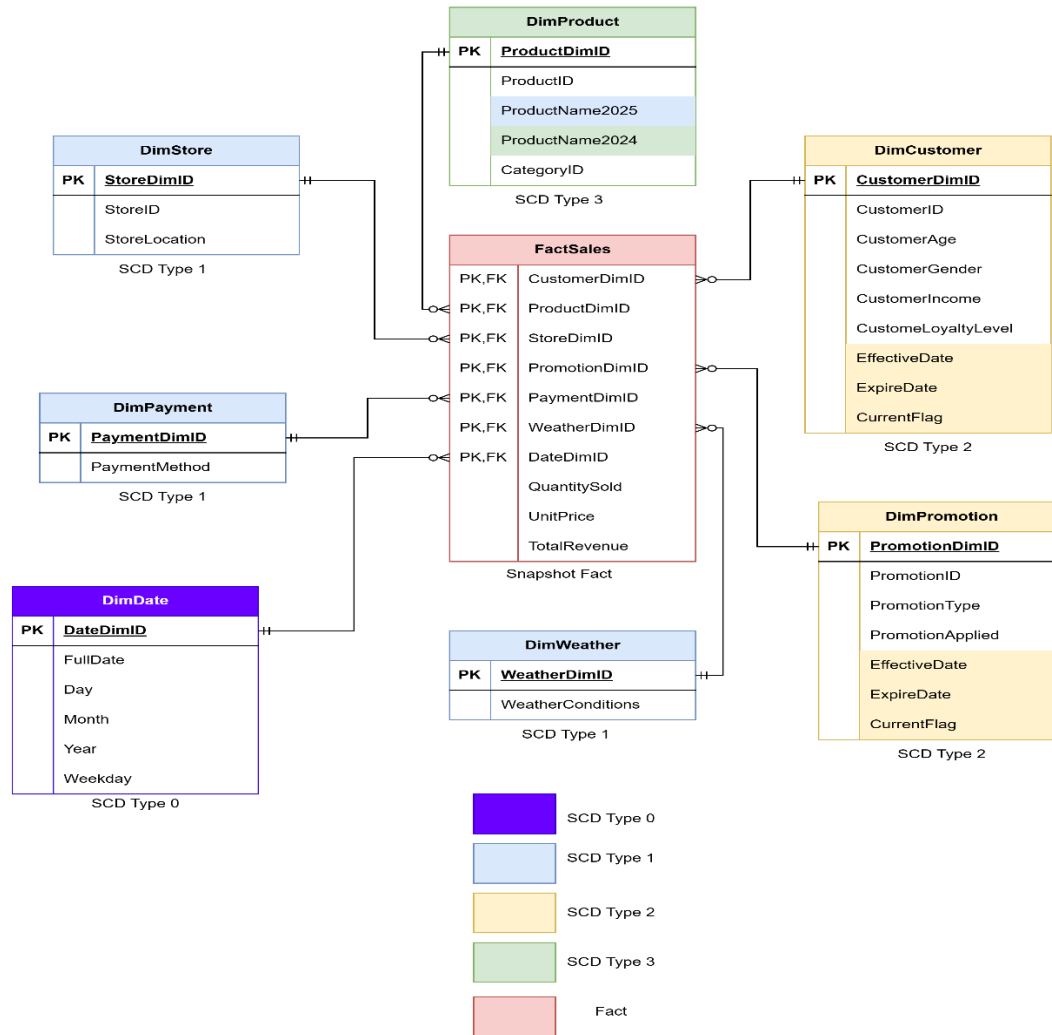
defined with appropriate primary and foreign keys to enforce referential integrity and support efficient joins. This normalization process not only ensured data consistency but also created a scalable foundation for subsequent OLAP modeling and analytical reporting.

### Entity-Relationship Diagram:



### 3) Star Model:

To enable efficient analytical querying and business reporting, the normalized OLTP schema was transformed into a dimensional **Star Schema**. This model centers around a primary **fact table** containing measurable business events—such as transactions—and connects to multiple descriptive **dimension tables** that provide context for analysis. This denormalized structure was optimized for OLAP workloads and used to power insights across sales, products, time, and store-level performance



### i) Dimensional Tables

The dimensional model consists of several descriptive tables that enrich transactional analysis. `dim_customer` stores demographic information such as age, gender, income, and loyalty level, and is implemented using SCD Type 2 to track historical changes in customer attributes over time. `dim_product` captures product details and uses SCD Type 3 to maintain both current and previous category information. `dim_store`, modeled with SCD Type 1, reflects only the latest store details. `dim_date` extracts structured components like day, month, and year from transaction timestamps. `dim_promotion` is also implemented using SCD Type 2, preserving historical variations in promotion types and their application logic. Additional dimensions such as `dim_payment_method` and `dim_weather` provide further transaction context. All dimension tables are linked to the central fact table via foreign keys, enabling robust, multi-dimensional analysis across time, customers, products, and promotions.

## ii) Fact Table

The central component of the star schema is the fact\_sales table, which stores measurable business events related to each transaction. This table captures foreign keys referencing all relevant dimension tables, including dim\_customer, dim\_product, dim\_store, dim\_date, dim\_promotion, dim\_payment\_method, and dim\_weather. Key measures in the fact table include quantity\_sold, unit\_price, actual\_demand, forecasted\_demand, and the stockout\_indicator, which is critical for supply chain and inventory analysis. By organizing transactional data in this structured format, the fact\_sales table enables fast, scalable analytical queries that support a wide range of business use cases—such as evaluating sales performance, analyzing promotion effectiveness, and forecasting demand patterns.

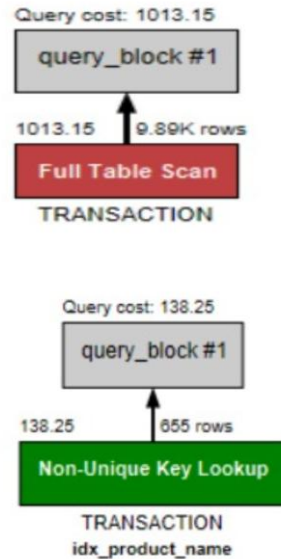
## 4) Stored Procedure:

To automate the transformation of OLTP data into the OLAP star schema, a stored procedure was developed to populate the fact\_sales table. This procedure reads cleaned and structured data from the transaction table and performs the necessary foreign key lookups to match each transaction to the appropriate dimension records. The procedure ensures accurate mapping by joining the OLTP transaction data with dimension tables such as dim\_customer, dim\_product, dim\_store, dim\_date, dim\_promotion, dim\_payment\_method, and dim\_weather using business keys (e.g., customer ID, product ID, transaction date). It then inserts the resulting records into fact\_sales, along with key performance metrics like quantity sold, unit price, forecasted demand, actual demand, and stockout indicator. By encapsulating this logic in a stored procedure, the ETL process becomes repeatable, efficient, and easier to maintain. This also reduces the risk of inconsistencies during fact table updates and improves overall system performance by executing logic directly within the database.

## 5) Performance Optimization

To improve query efficiency and reduce resource usage, indexing was applied to key columns in the OLTP schema. Specifically, an index named idx\_product\_name was created on the product\_id column in the transaction table. Before indexing, queries on this column resulted in a costly full table scan, as shown in the initial execution plan with over 9,800 rows scanned and a query cost exceeding 1000.

After indexing, the same query leveraged a non-unique key lookup, reducing the number of rows scanned to just 655 and lowering the query cost to 138. This significantly improved performance by minimizing I/O operations and buffer cache usage. Indexing thus played a critical role in optimizing transaction-level queries within the system.



Performance Tuning Example

## 6) SCD Type 2 and 3: Implementation with Examples

To manage changing business data while preserving historical accuracy, this project applies both SCD Type 2 and SCD Type 3 techniques across selected dimension tables. SCD Type 2 is used in dim\_customer and dim\_promotion to retain complete change history. For example, when a customer's income or loyalty level changes, a new record with a unique surrogate key is inserted into dim\_customer. Similarly, any change in promotion details results in a new row in dim\_promotion, enabling accurate tracking of which promotion was applied during a given transaction.

In contrast, SCD Type 3 is implemented in dim\_product to store both current and previous values of a changing attribute, specifically the product category. When a product's category changes, the current category value is shifted into a previous\_category field, and the new category is stored as the current one. This allows for limited historical comparison without adding new rows. Together, these techniques support richer trend analysis, customer tracking, and historical reporting.

customer_dim_id	customer_id	customer_age	customer_gender	customer_income	customer_loyalty_level	effective_date	expire_date	current_flag
25	1043	27	Other	95653.7	Bronze	2025-04-26	9999-12-31	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

customer_dim_id	customer_id	customer_age	customer_gender	customer_income	customer_loyalty_level	effective_date	expire_date	current_flag
25	1043	27	Other	95653.7	Bronze	2025-04-26	2025-04-26	0
8192	1043	27	Other	96653.7	Bronze	2025-04-26	9999-12-31	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

SCD TYPE 2 EXAMPLE

	product_dim_id	product_id	product_name_2024	product_name_2025	category_id
▶	1	1000	Fridge	NULL	1
*	NULL	NULL	NULL	NULL	NULL

	product_dim_id	product_id	product_name_2024	product_name_2025	category_id
▶	1	1000	Fridge	Frigidaire Fridge	1
*	NULL	NULL	NULL	NULL	NULL

### SCD TYPE 3 EXAMPLE

## 7) VIEWS AND TABLEAU

To support reporting and dashboard integration, SQL views were created to simplify access to analytical queries derived from the star schema. These views aggregate key measures such as total sales, quantity sold, and promotion usage across time, products, and locations. Each view was purpose-built to answer specific business questions and improve performance by reducing query complexity during visualization.

Using these views, interactive dashboards were developed in Tableau to address core business scenarios. These included tracking monthly sales by store location, evaluating the impact of promotions on sales volume, identifying the top 10 best-selling products, and pinpointing the highest-grossing store for a given month. The dashboards enable business users to gain clear, actionable insights from complex transactional data with minimal technical overhead.





Tableau Dashboard

### 3. Conclusion

This project successfully implemented a complete E-commerce Analytics pipeline using a Walmart retail dataset. The data was first normalized into a BCNF-compliant OLTP schema, then transformed into a dimensional star schema for OLAP analysis. SCD Types 1, 2, and 3 were applied to track evolving attributes across key dimensions.

A stored procedure automated the population of the fact\_sales table, while SQL views supported targeted business queries. Tableau dashboards provided clear insights into sales trends, promotions, and store performance. Overall, the project demonstrates an effective blend of data modeling, ETL, and business intelligence in a real-world context.