**Assignment 6**

**Introduction** The objective of this project is to design and implement a data warehouse for a customer order processing system for an enterprise with multiple stores across different cities and states. The data warehouse will integrate information from operational databases to support business intelligence and decision-making. The scope includes designing a schema, implementing data cubes, and enabling Online Analytical Processing (OLAP) functionalities such as roll-up, drill-down, slice, and dice.

**Business Requirements** The data warehouse system extracts data from the existing headquarter and sales databases to provide analytical insights to business users. The system should support queries related to store inventory, order fulfillment, customer order tracking, and headquarters' stock levels.

**Functional Specification**

* **Input:** Data from operational databases including customer information, orders, stores, and item stock levels.
* **Output:** Analytical reports and queries answering business questions related to inventory, customer orders, and headquarters' operations.

**Data Warehousing Design**

1. **Extract, Transform, Load (ETL) Process**
   * Extract data from headquarter and sales databases.
   * Transform data to conform to the warehouse schema.
   * Load the transformed data into the data warehouse.
2. **Schema Design – Star Schema**
   * **Fact Table:** Order\_Fact (Order\_no, Order\_date, Customer\_id, Store\_id, Item\_id, Quantity\_ordered, Ordered\_price)
   * **Dimension Tables:**
     + Customer\_Dim (Customer\_id, Customer\_name, City\_id, First\_order\_date, Customer\_type)
     + Store\_Dim (Store\_id, City\_id, Phone)
     + Item\_Dim (Item\_id, Description, Size, Weight, Unit\_price)
     + City\_Dim (City\_id, City\_name, State, Headquarter\_addr)

**Data Cube Implementation**

* The data warehouse will be implemented in MySQL.
* Data cubes will be created for analysis based on dimensions and measures.
* OLAP operations such as roll-up, drill-down, slice, and dice will be supported.

**Observations**

1. **OLAP Reports**
   * Query 1: Find all the stores along with city, state, phone, description, size, weight, and unit price that hold a particular item of stock.

SELECT s.Store\_id, c.City\_name, c.State, s.Phone, i.Description, i.Size, i.Weight, i.Unit\_price

FROM Stores s

JOIN City\_Dim c ON s.City\_id = c.City\_id

JOIN Stored\_items si ON s.Store\_id = si.Store\_id

JOIN Item\_Dim i ON si.Item\_id = i.Item\_id

WHERE i.Item\_id = ?;

Query 2: Find all the orders along with customer name and order date that can be fulfilled by a given store.

SELECT o.Order\_no, o.Order\_date, c.Customer\_name

FROM Order\_Fact o

JOIN Customer\_Dim c ON o.Customer\_id = c.Customer\_id

WHERE o.Store\_id = ?;

Query 3: Find all stores along with city name and phone that hold items ordered by a given customer.

SELECT DISTINCT s.Store\_id, c.City\_name, s.Phone

FROM Stores s

JOIN City\_Dim c ON s.City\_id = c.City\_id

JOIN Order\_Fact o ON s.Store\_id = o.Store\_id

WHERE o.Customer\_id = ?;

Query 4: Find the headquarter address along with city and state of all stores that hold stocks of an item above a particular level.

SELECT h.Headquarter\_addr, c.City\_name, c.State

FROM Headqarters h

JOIN City\_Dim c ON h.City\_id = c.City\_id

JOIN Stores s ON c.City\_id = s.City\_id

JOIN Stored\_items si ON s.Store\_id = si.Store\_id

WHERE si.Item\_id = ? AND si.Quantity\_held > ?;

Query 5: For each customer order, show the items ordered along with description, store id, and city name and the stores that hold the items.

SELECT o.Order\_no, i.Description, s.Store\_id, c.City\_name

FROM Order\_Fact o

JOIN Item\_Dim i ON o.Item\_id = i.Item\_id

JOIN Stores s ON o.Store\_id = s.Store\_id

JOIN City\_Dim c ON s.City\_id = c.City\_id;

Query 6: Find the city and the state in which a given customer lives.

SELECT c.City\_name, c.State

FROM Customer\_Dim cd

JOIN City\_Dim c ON cd.City\_id = c.City\_id

WHERE cd.Customer\_id = ?;

Query 7: Find the stock level of a particular item in all stores in a particular city.

SELECT s.Store\_id, si.Quantity\_held

FROM Stores s

JOIN Stored\_items si ON s.Store\_id = si.Store\_id

WHERE si.Item\_id = ? AND s.City\_id = ?;

Query 8: Find the items, quantity ordered, customer, store, and city of an order.

SELECT o.Order\_no, i.Description, o.Quantity\_ordered, c.Customer\_name, s.Store\_id, cd.City\_name

FROM Order\_Fact o

JOIN Item\_Dim i ON o.Item\_id = i.Item\_id

JOIN Customer\_Dim c ON o.Customer\_id = c.Customer\_id

JOIN Stores s ON o.Store\_id = s.Store\_id

JOIN City\_Dim cd ON s.City\_id = cd.City\_id;

Query 9: Find the walk-in customers, mail order customers, and dual customers (both walk-in and mail order).

SELECT c.Customer\_id, c.Customer\_name,

       CASE

           WHEN w.Customer\_id IS NOT NULL AND m.Customer\_id IS NOT NULL THEN 'Dual'

           WHEN w.Customer\_id IS NOT NULL THEN 'Walk-in'

           WHEN m.Customer\_id IS NOT NULL THEN 'Mail-order'

           ELSE 'Unknown'

       END AS Customer\_Type

FROM Customer\_Dim c

LEFT JOIN Walk-in\_customers w ON c.Customer\_id = w.Customer\_id

LEFT JOIN Mail\_order\_customers m ON c.Customer\_id = m.Customer\_id;