AY: 2025-26

Class:	**TE**	**Semester:**	**V**
Course Code:	**CSC504**	**Course Name:**	**Data Warehousing and Mining**

Name of Student:	**Shravani Sandeep Raut**
Roll No. :	**51**
Experiment No.:	**01**
Title of the Experiment:	**To Build a Data Warehouse – Star Schema, Snowflake Schema**
Date of Performance:	
Date of Submission:	

Evaluation

Performance Indicator	**Max. Marks**	**Marks Obtained**
Performance	5	
Understanding	5	
Journal work and timely submission	10	
Total	20	

Performance Indicator	**Exceed Expectations (EE)**	**Meet Expectations (ME)**	**Meet Expect Below Expectations (BE)**
Performance	4-5	2-3	1
Understanding	4-5	2-3	1
Journal work and timely submission	8-10	5-8	1-4

Checked by

Name of Faculty : Ms. Neha Raut
Signature :
Date:

Aim: To Build a Data Warehouse – Star Schema, Snowflake Schema and Fact Constellation Schema

Objective: A data warehouse is a large store of data collected from multiple sources within a business. The objective of a data warehouse system is to provide consolidated, flexible, meaningful data storage to the end user for reporting and analysis.

Theory:

In general, the warehouse design process consists of the following steps:

1. Choose a business process to model (e.g., orders, invoices, shipments, inventory, account administration, sales, or the general ledger). If the business process is organizational and involves multiple complex object collections, a data warehouse model should be followed. However, if the process is departmental and focuses on the analysis of one kind of business process, a data mart model should be chosen.
2. Choose the business process grain, which is the fundamental, atomic level of data to be represented in the fact table for this process (e.g., individual transactions, individual daily snapshots, and so on).
3. Choose the dimensions that will apply to each fact table record. Typical dimensions are time, item, customer, supplier, warehouse, transaction type, and status.
4. Choose the measures that will populate each fact table record. Typical measures are numeric additive quantities like dollars sold and units sold.

Steps to Draw Star, Snowflake, and Fact Constellation Schemas

1. Star Schema:

- Step 1: Identify the central fact table, which contains quantitative data (e.g., sales, revenue).
- Step 2: Determine the dimension tables related to the fact table, such as time, product, customer, etc.
- Step 3: Define the relationships between the fact table and each dimension table, usually a one-to-many relationship.
- Step 4: Draw the fact table at the center and connect it to each dimension table using lines, creating a star-like structure.

2. Snowflake Schema:

- Step 1: Start with the fact table as in the Star Schema.

- Step 2: Identify the dimension tables and further normalize them by breaking them into multiple related tables (e.g., split "Location" into "Country" and "City").
- Step 3: Establish relationships between these normalized dimension tables and the fact table.
- Step 4: Draw the fact table at the center, then connect it to the dimension tables, which in turn connect to their sub-tables, forming a snowflake-like structure.

3. Fact Constellation Schema:

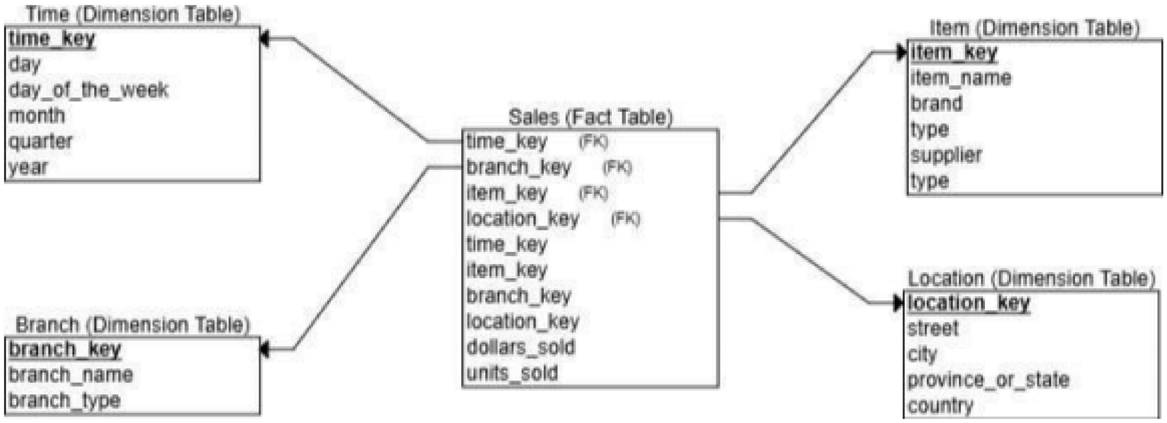
- Step 1: Identify multiple fact tables representing different processes or subjects (e.g., sales and inventory).
- Step 2: Identify the shared dimension tables that will connect to these fact tables.
- Step 3: Define relationships between each fact table and the shared dimension tables.
- Step 4: Draw all fact tables and connect them to the shared dimension tables, creating a constellation of facts and dimensions.

Problem Statement:

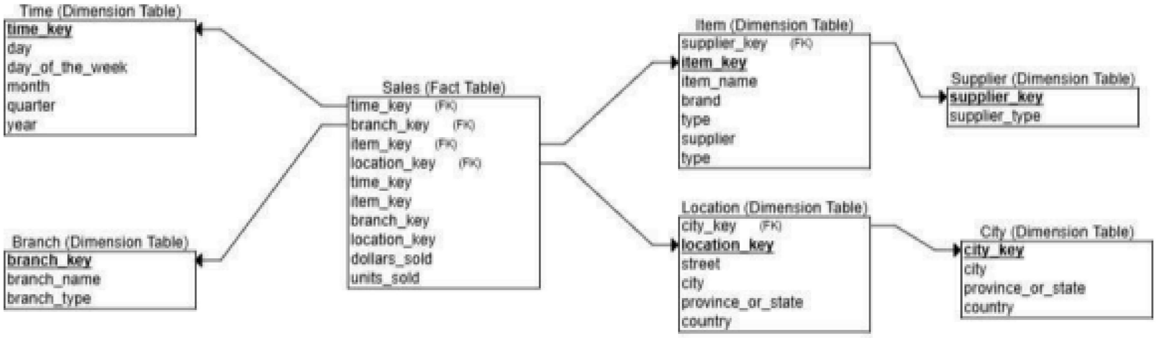
The problem is to design and implement a data warehousing solution for a bookstore that optimally organizes and manages its vast data, including sales, inventory, customer information, and more, to facilitate efficient reporting and analytics. This involves creating both a star schema and a snowflake schema to support various business intelligence and decision-making processes, while ensuring data accuracy, integrity, and performance

Construction of Star schema and Snowflake schema:

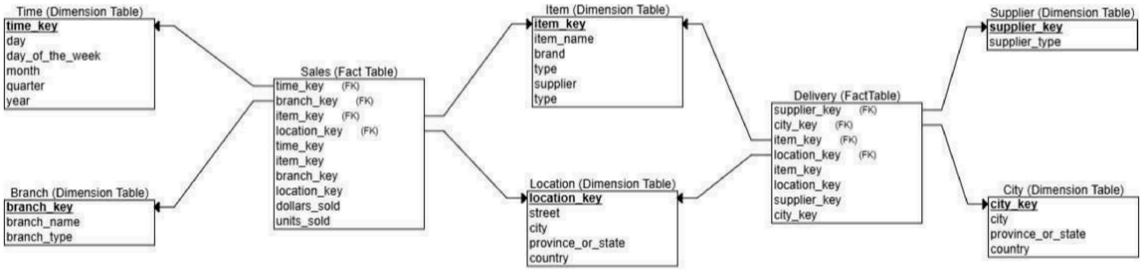
Star schema:



Snowflake schema:



Fact Constellation Schemas



Conclusion:

1. How does the Snowflake Schema compare to the Star Schema in terms of ease of maintenance and scalability?

   The Star Schema is easier to maintain due to its simple design and fewer joins, making it ideal for read-heavy operations and straightforward scalability. In contrast, the Snowflake Schema, with its normalized structure, is more complex to maintain but better suited for handling diverse datasets and ensuring data integrity. The choice depends on the specific needs of the data warehousing environment.

2. How do you manage the complexity of ETL (Extract, Transform, Load) processes in a Fact Constellation Schema?

   Modular ETL Design
   Break down ETL workflows into smaller, reusable modules or components.Create separate pipelines for each fact table and dimension table but reuse dimension loading processes where possible.This modularity simplifies troubleshooting and maintenance.

   Consistent Dimension Handling
   Since multiple fact tables share dimensions, ensure dimension data is consistent and up to-date.Use Slowly Changing Dimension (SCD) strategies carefully to maintain dimension history without affecting multiple fact tables.Centralize dimension loading and updates to avoid redundant processes.

   Data Validation and Quality Checks
   Implement validation checks at each ETL stage to catch errors early.Validate data consistency across shared dimensions and fact tables to prevent mismatches.

   Automation and Orchestration
   Use ETL orchestration tools (e.g., Apache Airflow, Azure Data Factory) to schedule and

monitor workflows.Automate dependencies between dimension and fact loads to ensure correct
sequence.

Incremental Loading
Optimize ETL by loading only new or changed data rather than full refreshes.Incremental loads reduce processing time and complexity, especially with multiple fact tables.

Metadata Management
Maintain clear metadata documentation for fact and dimension tables, including relationships.Helps ETL developers understand dependencies and data lineage.

Testing and Version Control
Implement rigorous testing for each ETL component.Use version control systems to track ETL script changes, facilitating rollback and collaboration.