

Experiment – 1 Build Data Warehouse/Data Mart for a given problem statement

- i) Identifying the source tables and populating sample data**
- ii) Design dimensional data model i.e. Star, Snowflake, and Fact-Constellation schema**

Name: Shaikh Shadab **Rollno** : 17DCO74
Class : TE.CO **Batch** : B3

#Theory

➤ Schemas, Dimensions, Facts

The database schema of a database system is its structure described in a formal language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases).

A "fact" is a numeric value that a business wishes to count or sum. A "dimension" is essentially an entry point for getting at the facts. Dimensions are things of interest to the business

➤ Dimension Modelling

A Dimensional Model is a database structure that is optimized for online queries and Data Warehousing tools. It is comprised of "fact" and "dimension" tables. Dimensional Models are designed for reading, summarizing and analyzing numeric information

➤ Dimension and Fact tables

A dimension table stores attributes, or dimensions, that describe the objects in a fact table. Companion table to the fact table contains descriptive attributes to be used as query constraining. Design Should be wordy, descriptive, complete, and quality assured.

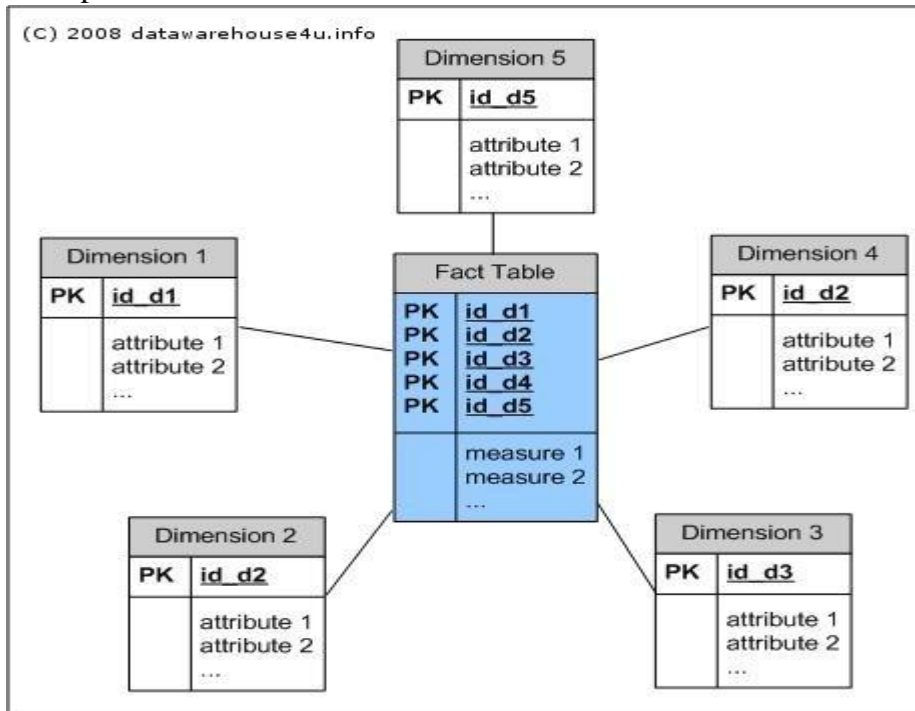
A fact table stores quantitative information for analysis and is often denormalized. A fact table works with dimension tables and it holds the data to be analysed and a dimension table stores data about the ways in which the data can be analysed. Thus, a fact table consists of two types of columns. The foreign keys column allows to join with dimension tables and the measure columns contain the data that is being analysed.

Design is defined by their grain or its most atomic level.

➤ Star Schema

Star schema architecture is the simplest data warehouse schema. It is called a star schema because the diagram resembles a star, with points radiating from a centre. The centre of the star consists of fact table and the points of the star are the dimension tables. Usually the fact tables in a star schema are in third normal form(3NF) whereas dimensional tables are de-normalized. Despite the fact that the star schema is the simplest architecture, it is most commonly used nowadays and is recommended by Oracle.

Example of star schema:

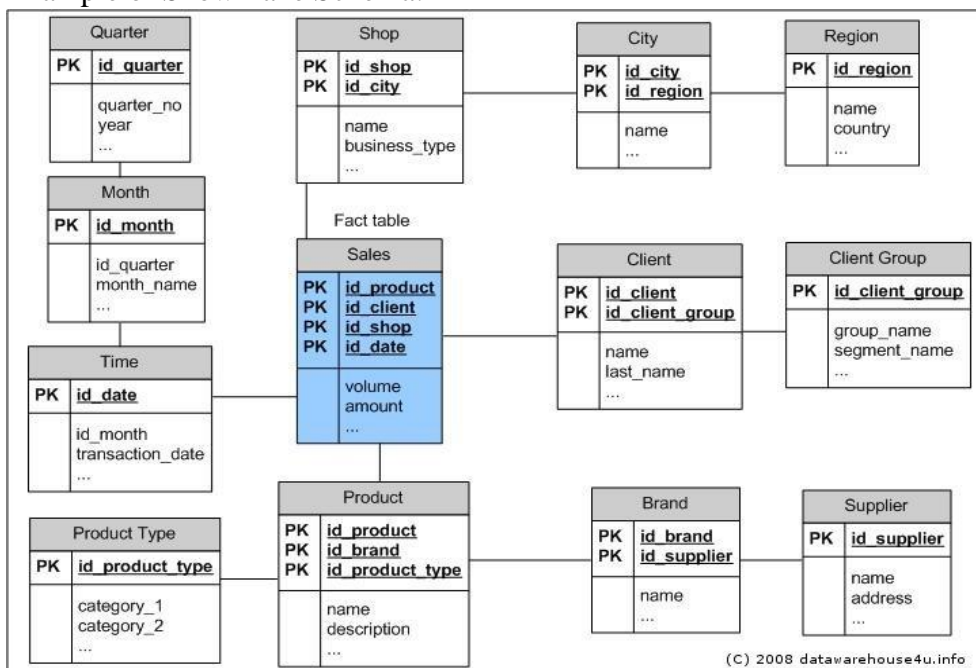


➤ Snowflake Schema

“Snowflaking” is a method of normalizing the dimension tables in a STAR schema. When we completely normalize all the dimension tables up to 3NF, the resultant structure resembles a snowflake with the fact table in the middle.

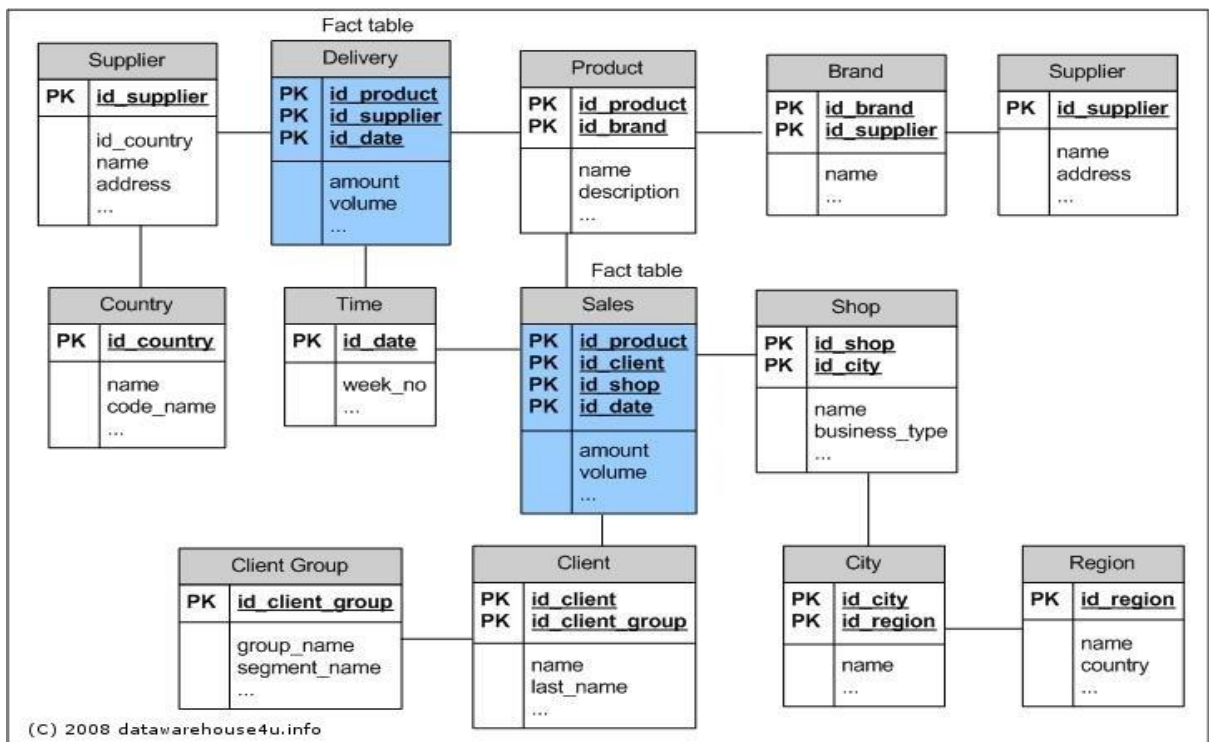
The snowflake schema architecture is a more complex variation of the star schema used in a data warehouse, because the tables which describe the dimensions are normalized.

Example of Snowflake Schema:



➤ **Fact constellation schema**

Also known as galaxy schema, for each star schema it is possible to construct fact constellation schema (for example by splitting the original star schema into more star schemes each of them describes facts on another level of dimension hierarchies). The fact constellation architecture contains multiple fact tables that share many dimension tables. The main shortcoming of the fact constellation schema is a more complicated design because many variants for particular kinds of aggregation must be considered and selected. Moreover, dimension tables are still large. It can be seen as an extension of the star schema. A collection of multiple fact tables sharing dimension tables, viewed as a collection of stars.



➤ **Problem definition used in this experiment**

Design a star schema to track a shipments for a distribution company , the following dimension table are found

- i. Time
- ii. Cust_Shipto
- iii. ShipFrom
- iv. Product
- v. TypeOfDeal
- vi. ModeOfShipment

Review this dimension and list possible attributes for each of dimension table also designate a primary key for each table

i) **Identifying the source tables and populating sample data**
#Inserting data in product dimension

Dim_Product - Table

Table Name: Dim_Product Schema: mydb

Filter Rows: Edit: Export/Import: Wrap Cell Content: Apply changes:

ProductID	Name	Code	ModelNumber	Desc
1001	PulseCandyJar	M0163A	10012021000219	Raw Mango with tangy twist
NULL	NULL	NULL	NULL	NULL

#Inserting data in time dimension

Dim_Time - Table

Table Name: Dim_Time Schema: mydb

Filter Rows: Edit: Export/Import: Wrap Cell Content: Apply changes:

TimeID	DispatchTime	ReceivingTime	WeekNumber	Month	Year
Time1001	2019-02-12 12:00:00	2019-02-14 12:00:00	3	Feb	2019
NULL	NULL	NULL	NULL	NULL	NULL

#Forward engineering wizard screen initial screen

Forward Engineer to Database

Connection Options

Select Objects

Review SQL Script

Commit Progress

Set Parameters for Connecting to a DBMS

Stored Connection: Local instance MySQL Router Select from saved connection settings

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname: localhost Port: 3306 Name or IP address of the server host - and TCP/IP port.

Username: root Name of the user to connect with.

Password: Store in Vault ... Clear The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

Back Next Cancel

#Wizard screen 2 ticking on generate insert statements

The screenshot shows the 'Forward Engineer to Database' wizard, screen 2, titled 'Set Options for Database to be Created'. The left sidebar contains a tree view with 'Options' selected. The main area is divided into three sections: 'Tables', 'Other Objects', and 'Code Generation'. In the 'Tables' section, the checkbox 'Generate INSERT statements for tables' is checked. In the 'Code Generation' section, the checkbox 'Include model attached scripts' is checked. At the bottom right are 'Back', 'Next', and 'Cancel' buttons.

Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Set Options for Database to be Created

Tables

- ☐ Skip creation of FOREIGN KEYS
- ☐ Skip creation of FK Indexes as well
- ☐ Generate separate CREATE INDEX statements
- ☒ Generate INSERT statements for tables
- ☐ Disable FK checks for INSERTs

Other Objects

- ☐ Don't create view placeholder tables
- ☒ Do not create users. Only create privileges (GRANTS)

Code Generation

- ☐ DROP objects before each CREATE object
- ☐ Generate DROP SCHEMA
- ☐ Omit schema qualifier in object names
- ☐ Generate USE statements
- ☐ Add SHOW WARNINGS after every DDL statement
- ☒ Include model attached scripts

Back Next Cancel

#Wizard screen 3 ticking on export MySQL Table objects

The screenshot shows the 'Forward Engineer to Database' wizard, screen 3, titled 'Select Objects to Forward Engineer'. The left sidebar contains a tree view with 'Select Objects' selected. The main area contains a list of object types with checkboxes and 'Show Filter' buttons. The 'Export MySQL Table Objects' checkbox is checked, and it shows '9 Total Objects, 9 Selected'. Other object types are unchecked and show '0 Total Objects, 0 Selected'. At the bottom right are 'Back', 'Next', and 'Cancel' buttons.

Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

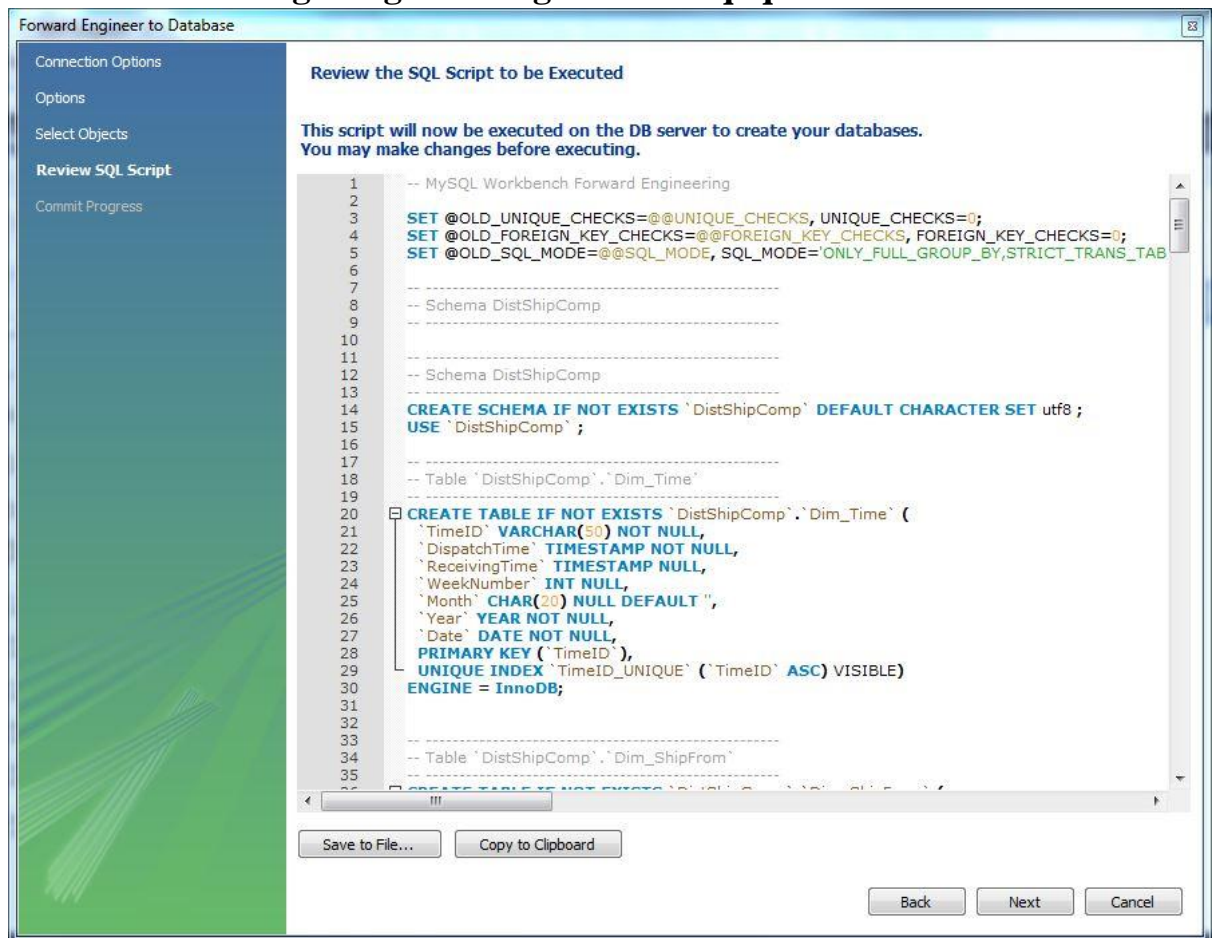
Select Objects to Forward Engineer

To exclude objects of a specific type from the SQL Export, disable the corresponding checkbox. Press Show Filter and add objects or patterns to the ignore list to exclude them from the export.

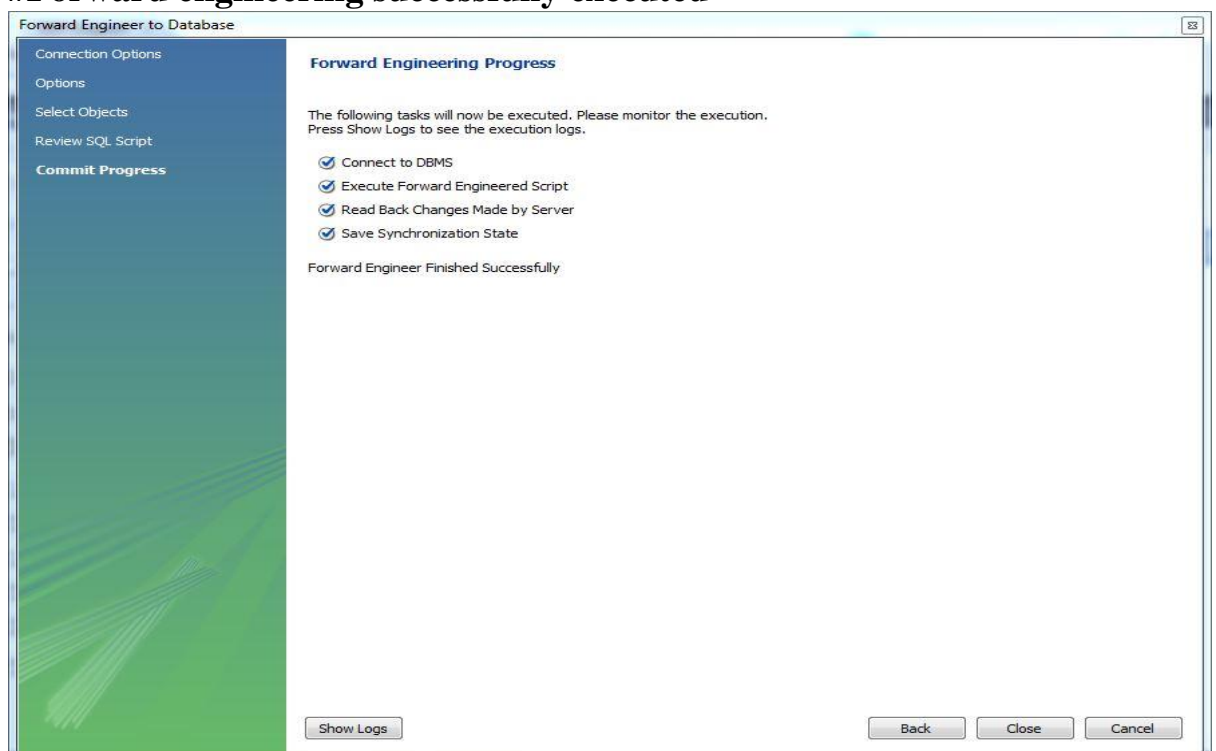
- ☒ Export MySQL Table Objects
9 Total Objects, 9 Selected Show Filter
- ☐ Export MySQL View Objects
0 Total Objects, 0 Selected Show Filter
- ☐ Export MySQL Routine Objects
0 Total Objects, 0 Selected Show Filter
- ☐ Export MySQL Trigger Objects
0 Total Objects, 0 Selected Show Filter
- ☐ Export User Objects
0 Total Objects, 0 Selected Show Filter

Back Next Cancel

#Wizard screen 4 getting a list of generated sql queries and statements

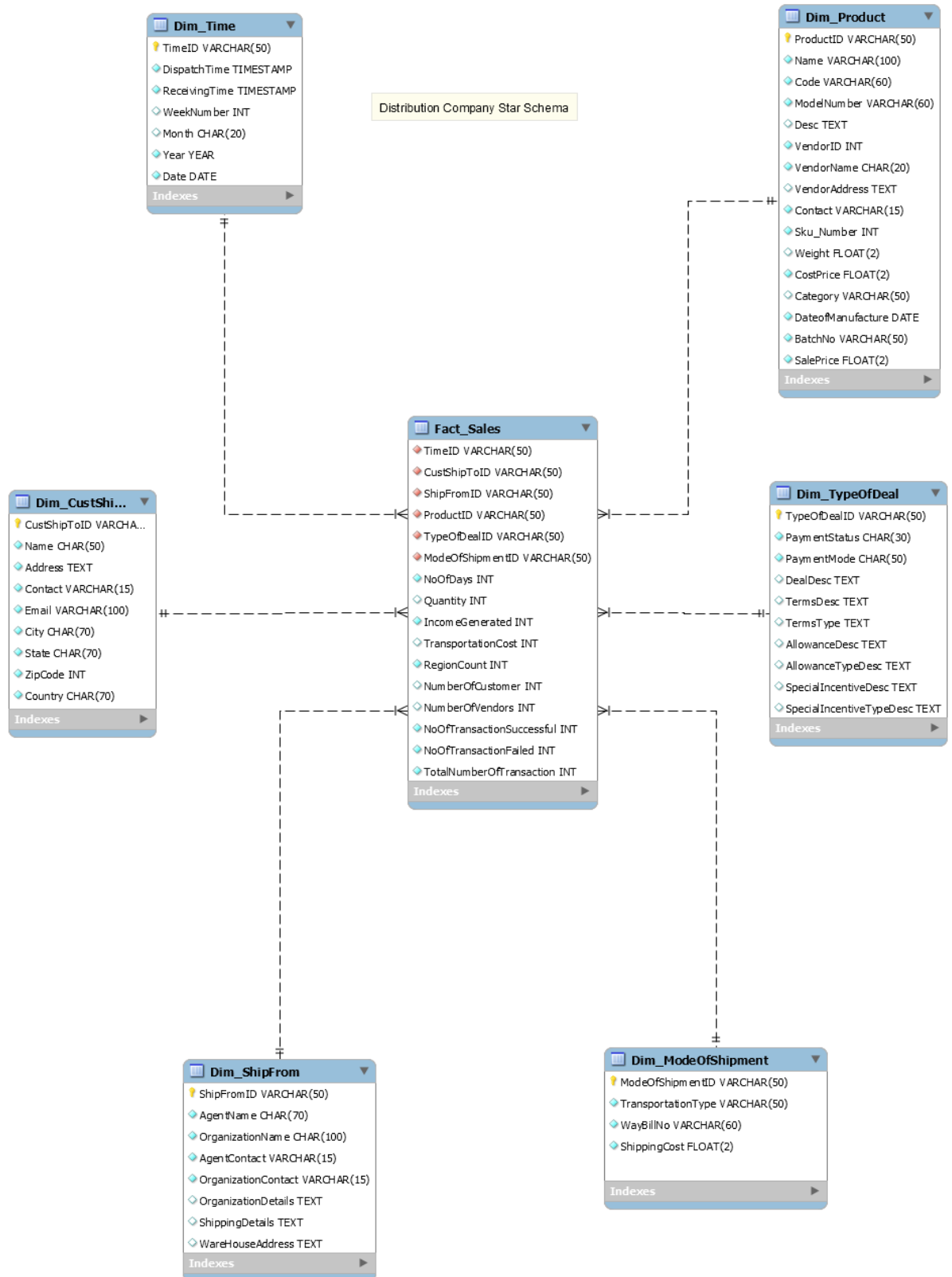


#Forward engineering successfully executed

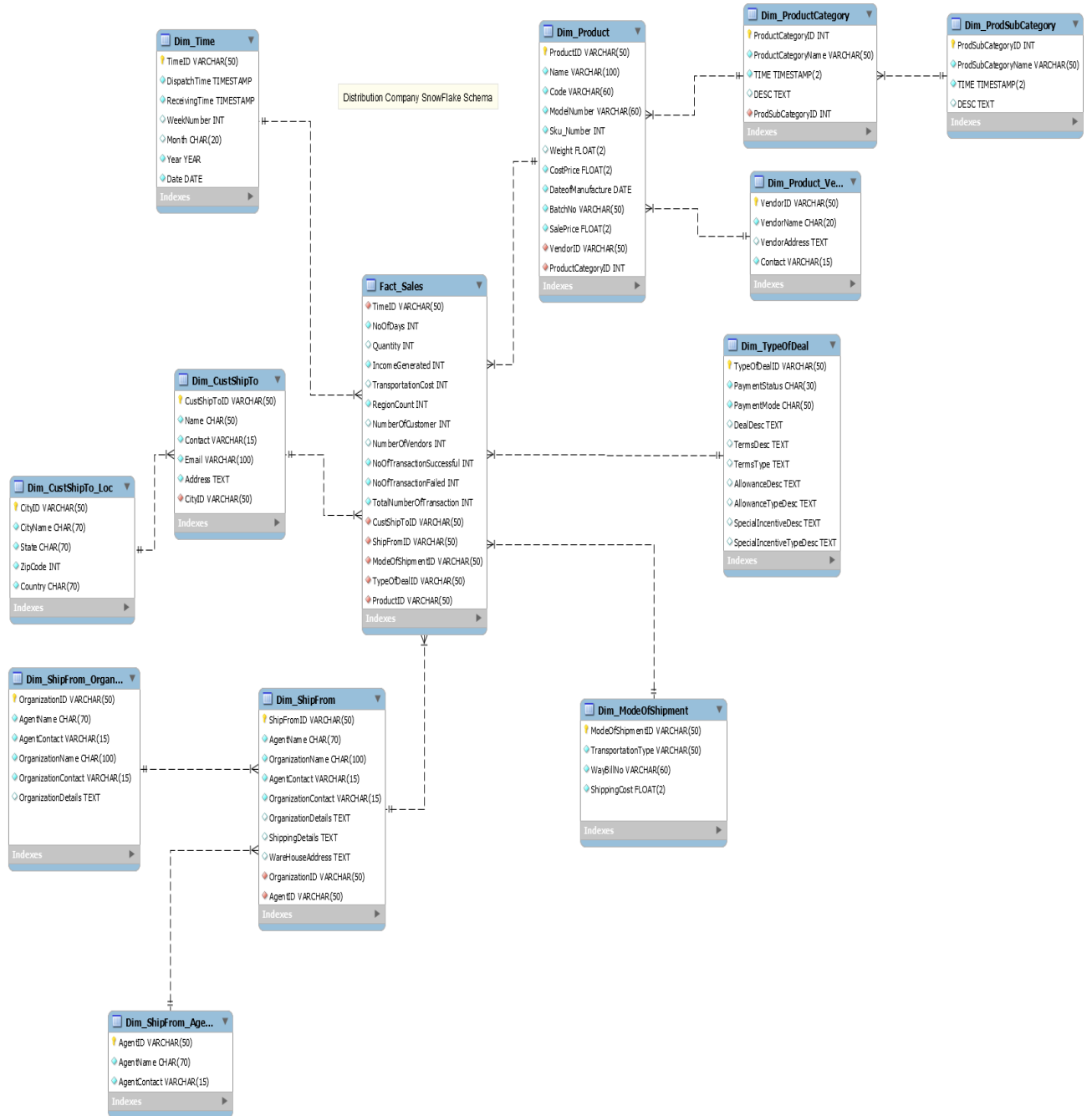


ii) Design dimensional data model i.e. Star, Snowflake, and Fact-Constellation schema

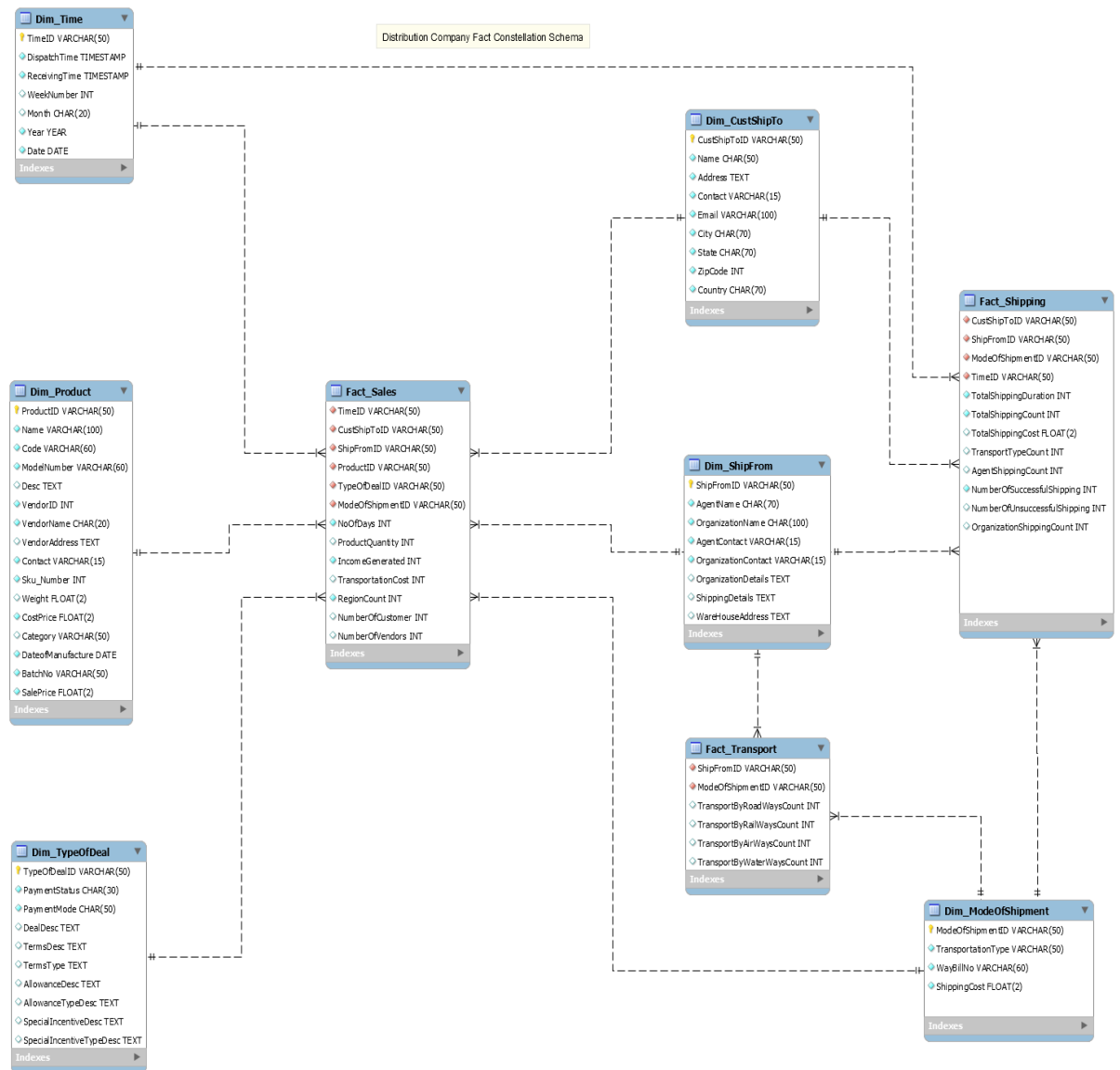
#Star Schema for distribution company with proposed dimension



#Snowflake schema for the distribution company with proposed dimension in normalized form



#FactConstellation (Galaxy) schema for the distribution company with proposed dimension and more than 1 fact tables.



#Conclusion

From i) part of this experiment we learned the basic concept about dimension modelling and were introduced with new data mart and Datawarehouse schema designing technique we have populated the sample data and shown each step prompted by wizard

From ii) part of the experiment we have implemented the given problem definition of a shipment company which ships and track products in all the three schemas first, star then we normalized it to create snowflake and, in the end, we build multiple fact tables resulting into galaxy schema