Mapping and comparison of data models in data warehouse

Jaya Gupta

University School of Information Technology, GSSIPU
Delhi, India
jaya_gupta8@yahoo.com

***Abstract*— Many organizations owns billions of bytes data but may suffer from problem to access and analyze data. In order to solve this problem new concept and tools have evolved into a new information technology called data warehousing. A Data Warehouse is a collection of technologies aimed at enabling the knowledge worker to make better and faster decisions. Data warehouse is designed by using data models which act as a blueprint for the information requirements of an organization. Without a data model, the understanding, implementation and maintenance of the data warehouse are difficult. There are three levels of abstractions in the data modeling i.e. conceptual, logical and physical. Here in this paper we will create these data models and map one model from the earlier model which greatly helps in designing a data warehouse.**

Keywords— **Data warehouse, Conceptual Modeling, Logical Modeling, Physical Modeling**

I. INTRODUCTION

A data warehouse consists of a computer database responsible for the collection and storage of information for any organization [2]. This collection of information is then used to manage information efficiently and analyze the collected data. Data warehouses have become the key trend in corporate computing in the last years, since they provide managers with the most accurate and relevant information to improve strategic decisions. Data models are required in order to build data warehouses. Data modeling is divided into three levels i.e. conceptual, logical, and physical [5]. Complexity increases from conceptual to logical to physical. The conceptual model is concerned with the real world view and understanding of data, the logical model is a generalized formal structure in the rules of information science and ensures that all business requirements, definitions, and rules are supported, the physical model specifies how this will be executed in a particular DBMS instance and ensures optimal performance in the planning of indexes, relationships, data types, and properties. To support developers of OLAP, data-mining, and reporting systems, the data model also acts as documentation for the final data warehouse.

II. DATA MODELING IN DATA WAREHOUSE

Data models are used to design the data warehouse. The major factors to consider in data warehouse database design are: data size and complexity; query composition and complexity; query load; and query concurrency. Evaluation of these factors will result in different data models in data warehouse. Modeling data warehouses presents new data design challenge which greatly enhance decision making [1].

To start with data modeling firstly the requirements for any organizations is gathered. Then we transform those requirements into the conceptual model. Then we map the conceptual model to conceptual model and finally the physical model is created.

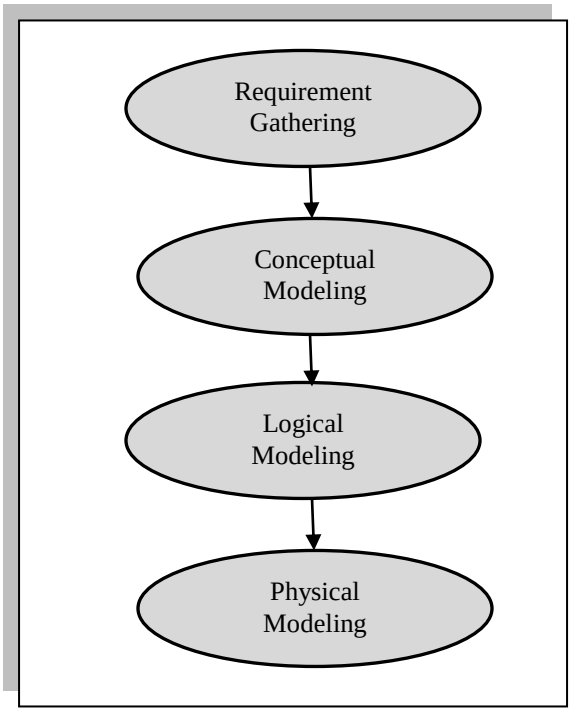

Figure 1: Data Modeling

A. CONCEPTUAL MODELING

Conceptual schema represents the semantics of an organization and describes the things of significance to an organization (classes), about which it is inclined to collect information, and characteristics of (attributes) and associations between pairs of those things of significance (relationships) [7].

B. CASE STUDY

We have taken a case study of a data warehouse conceptual model. In this, we are interested in analyzing the mobile store sales (Fact Mobile_Sales) of a big showroom.

This Fact contains the specific measures to be analyzed, i.e., quantity and price. On the other hand, the main dimensions along with we would like to analyze these measures are the Time they were sold, the specific model sold, the Customer to whom they were sold and the sales representative who have sold the model. Finally, Base classes Week, Quarter and Year; and City and Country represent the classification hierarchies of the Time and Customer dimensions, respectively, along with we are interested in analyzing measures. Fact classes are defined as compound classes in a shared aggregation relationship of n dimension classes.

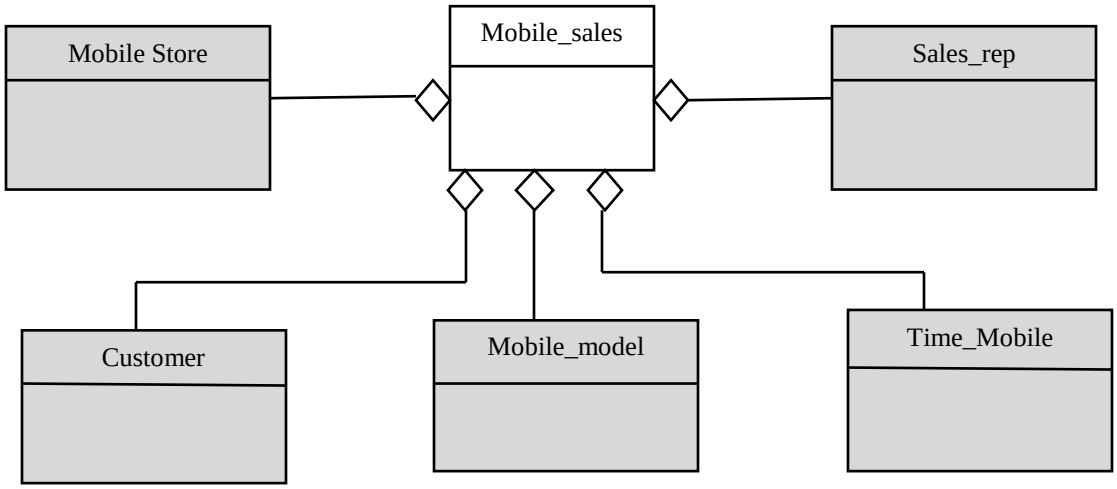

Figure 2: Conceptual Model

Hence the conceptual model for the mobile store is created. Now the next step is to map conceptual model to logical model.

C. LOGICAL MODEL

Logical model describes the structure of some domain of information [6]. This consists of descriptions of tables and columns, object oriented classes, and XML tags, among other things. Logical Model facilitates avoidance of data redundancy and thus prevents data & business transaction inconsistency. It Facilitates data re-use and sharing and decreases development and maintenance time and cost.

Features of a logical data model include:

- Includes all entities and relationships among them.
- All attributes for each entity are specified.
- The primary key for each entity is specified.
- Foreign keys (keys identifying the relationship between different entities) are specified.
- Normalization occurs at this level.

The steps for designing the logical data model are as follows:

1. Specify primary keys for all entities.
2. Find the relationships between different entities.
3. Find all attributes for each entity.
4. Resolve many-to-many relationships.
5. Normalization.

The figure below is an example of a logical data model which is mapped from the above case study.

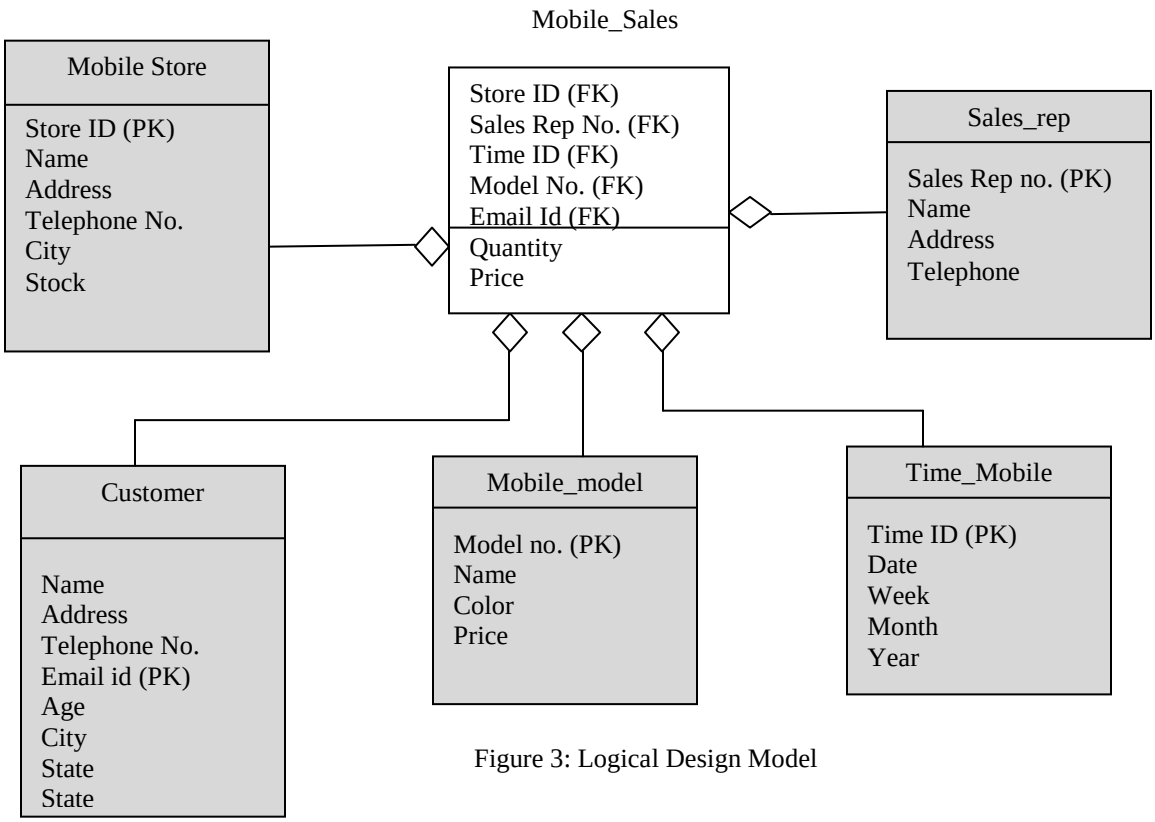

Figure 3: Logical Design Model

Comparing the logical data model shown above with the conceptual data model diagram, we see the main differences between the two:

- In a logical data model, primary keys are present, whereas in a conceptual data model, no primary key is present.
- In a logical data model, all attributes are specified within an entity. No attributes are specified in a conceptual data model.
- Relationships between entities are specified using primary keys and foreign keys in a logical data model. In a conceptual data model, the relationships are simply stated, not specified, so we simply know that two entities are related, but we do not specify what attributes are used for this relationship.

D. PHYSICAL DESIGN

Physical data model represents how the model will be built in the database. A physical database model shows all table structures, including column name, column data type, column constraints, primary key, foreign key, and relationships between tables. Features of a physical data model include:

- Specification all tables and columns.
- Foreign keys are used to identify relationships between tables.
- Denormalization may occur based on user requirements.
- Physical considerations may cause the physical data model to be quite different from the logical data model.

Physical data model will be different for different RDBMS. For example, data type for a column may be different between MySQL and SQL Server. The steps for physical data model design are as follows:

1. Convert entities into tables.
2. Convert relationships into foreign keys.
3. Convert attributes into columns.
4. Modify the physical data model based on physical constraints / requirements.

The figure below is an example of a physical data model which is mapped from logical model.

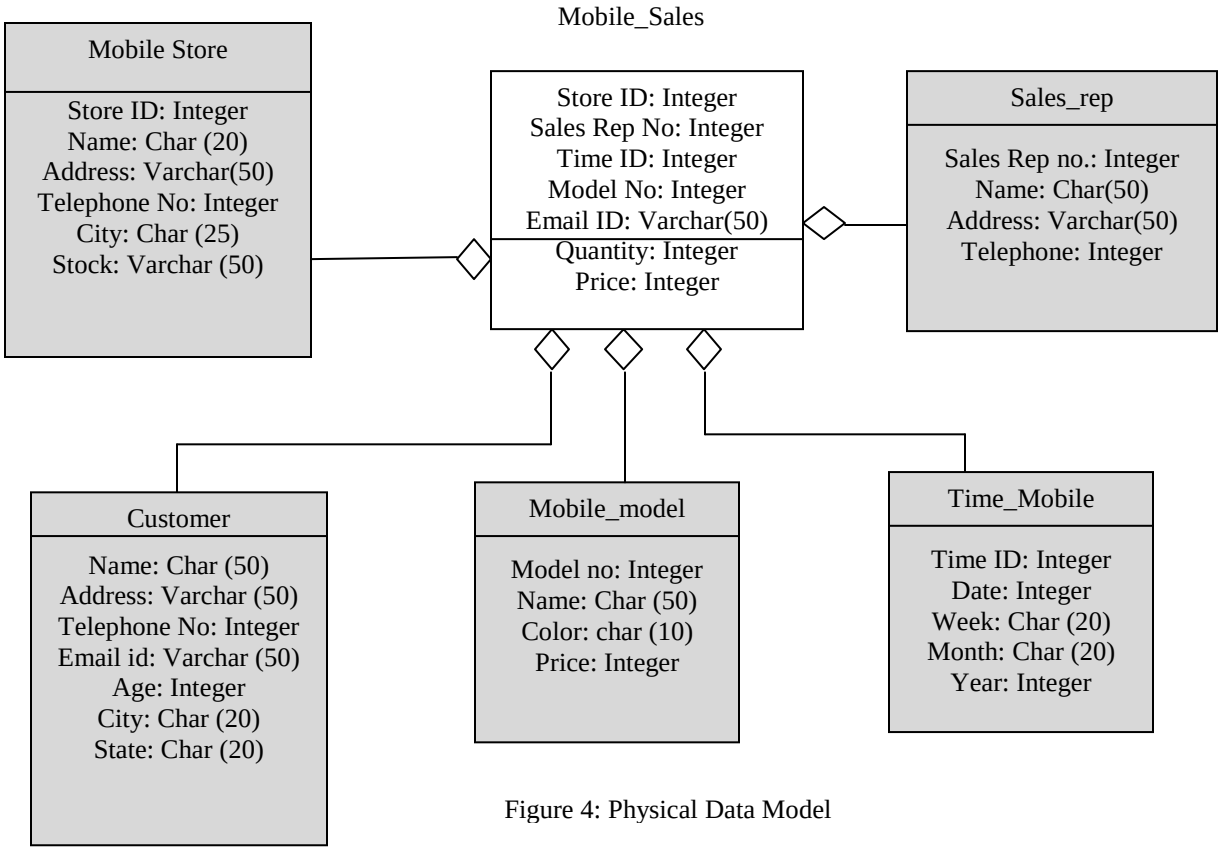

Figure 4: Physical Data Model

Comparing the logical data model shown above with the logical data model diagram, we see the main differences between the two:

- Entity names are now table names.
- Attributes are now column names.
- Data type for each column is specified. Data types can be different depending on the actual database being used.

Hence these are the various data models used to design data warehouse.

III. COMPARING FEATURES OF DATA MODELS

Various features are compared below in the table as shown:

Table 1: Comparison of models based on their features

FEATURES	CONCEPTUAL MODEL	LOGICAL MODEL	PHYSICAL MODEL
❖ ENTITIES	✓	✓	✓
❖ RELATIONSHIP	✓	✓	✓
❖ ATTRIBUTES	×	✓	✓
❖ PRIMARY KEY	×	✓	✓
❖ FOREIGN KEY	×	✓	✓
❖ NORMALISATION	×	✓	×
❖ TABLES	×	×	✓
❖ COLUMNS	×	×	✓

❖ DATA TYPES	×	×	✓
❖ TRIGGERS	×	×	✓
❖ PROCEDURES	×	×	✓
❖ ACCESS CONSTRAINTS	×	×	✓
❖ DENORMALISATION	×	×	✓
❖ DOMAINS	×	×	✓

IV. Conclusions

Data warehouse is the central repository which contains information for decision making. Designing data warehouse is the complex task. In order to design data warehouse various data models have been defined. They are conceptual, logical and physical and logical. A conceptual entity-relationship model shows how the business world sees information. It suppresses non-critical details in order to emphasize business rules and user objects. For a logical data model to be normalized it must include the full population of attributes to be implemented and those attributes must be defined in terms of their domains or logical data types.

The physical data model specifies implementation details which may be features of a particular product or version, as well as configuration choices for that database instance. In this paper we have taken a case study of mobile sales and created conceptual model and mapped the logical and physical model from it.

References

[1] J. Trujillo, M. Palomar, J. Go´mez, I.-Y. Song (2001) Designing Data Warehouses with OO Conceptual Models. IEEE Computer, Special issue on Data Warehouses 34, 66–75.

[2] Mario Piattini, Ismael Caballero, Marcela Genero, Coral Calero (1999) Data Quality and Database Design.

[3] Sergio Luján-Mora (2001) Multidimensional Modeling using UML and XML, Universidad de Alicante, Spain.

[4] Panos Vassiliadis, Mokrane Bouzeghoub, Christoph Quix (1999) Towards Quality-Oriented Data Warehouse Usage and Evolution. CAiSE'99, LNCS 1626, pp. 164-179.

[5] Sergio Luján Mora, Juan Trujillo(2004) Physical Modeling of Data Warehouses using UML, *DOLAP'04,* Washington, DC, USA.

[6] Verónika Peralta, Data Warehouse Logical Design from Multidimensional Conceptual Schemas, Universidad de la República, Uruguay.

[7] Moody Daniel L (2005) Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions, Data and Knowledge Engineering ,Volume 44,Issue 3,pp-243-276.

[8] Gema Berenguer, Rafael Romero, Juan Trujillo, Manuel Serrano, and Mario Piattini (2005). A Set of Quality Indicators and Their Corresponding Metrics for Conceptual Models of Data Warehouses. Springer, LNCS 3589, pp. 95 – 104.

[9] S. Luján-Mora, P. Vassiliadis, and J. Trujillo (2004). Data Mapping Diagrams for Data Warehouse Design with UML. In Proceedings of the 23rd International Conference on Conceptual Modeling (ER'04), Lecture Notes in Computer Science, Shanghai, China, Springer-Verlag.

[10] S. Luján-Mora and J. Trujillo (2003). A Comprehensive Method for Data Warehouse Design. In Proceedings of the 5th International Workshop on Design and Management of Data Warehouses (DMDW'03).