Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.74 – 79

RESEARCH ARTICLE

Data Mart Designing and Integration Approaches

Rashmi Chhabra[1], Payal Pahwa[2]

[1]Research Scholar, CSE Department, NIMS University, Jaipur (Rajasthan), India

[2]Bhagwan Parshuram Institute of Technology, I.P.University, Delhi, India

[1] rashmidahra@gmail.com, [2] pahwapayal@gmail.com

Abstract— Today companies need strategic information to counter fiercer competition, extend market share and improve profitability. So they need information system that is subject oriented, integrated, non volatile and time variant. Data warehouse is the viable solution. It is integrated repository of data gathered from many sources and used by the entire enterprise. In order to standardize data analysis and enable simplified usage patterns, data warehouses are normally organized as problem driven, small units called data marts. Each data mart is dedicated to the study of a specific problem. The data marts are merged to create data warehouse. This paper discusses about design and integration of data marts and various techniques used for integrating data marts.

Keywords - Data Warehouse; Data Mart; Star schema; Multidimensional Model; Data Integration

---

I. INTRODUCTION

A data warehouse is a subject-oriented, integrated, time variant, non-volatile collection of data in support of management's decision-making process [1]. Most of the organizations these days rely heavily on the information stored in their data warehouse. In companies, different departments develop their data marts independently. They develop the data marts according to their departmental needs. Finance has its data mart, marketing has its own, sales have theirs and so on. An individual data mart works well, when the data for a particular department is required. But for decision making purposes, an enterprise may require inter departmental data as well. Thus the need for integrating data marts arises. The task of integrating the different data marts is not as easy as they are developed in different environment. Each data mart represent the view of a single business process on integrating them we get the entire view of organization. For making decision for the organization integration is important as well as required. The layout of paper is as follows: Section II discusses Multidimensional Model and its representation, Data Mart and reasons for developing data marts are discussed in Section III, Section IV discusses approaches for designing data marts. Section V discusses the data integration and its approaches and the Section VI concluded our work.

II. MULTIDIMENSIONAL MODEL

The n-dimensional view of data is modeled using a multidimensional model[2]. A multidimensional data model is typically organized around a central theme transaction. The basic components of a multidimensional model are fact and dimensions. Fact table consist of the measurement, matrices or facts of a business process. It provide the additive values that act as independent variables by which dimensional attributes are analyzed. The facts are numerical measures like quantity, amount etc. . In fig. 1 Sales fact table quantity sold, sales amount and cost amount are the facts. Facts are usually quantities, which are used for analyzing relationship between dimensions. Dimensions are the perspective entities with respect to which organizations would like to keep records [6]. For example a Bank may create a customer warehouse in order to keep records of the customers with respect to the dimensions namely time, transaction, branch and location. Each dimension may have a table associated with it, called the dimension table. There are two basic approaches to multidimensional modeling one is star schema and other is cube [10]. Conceptually a MDB uses the idea of data cube to represent the dimensions of data available to a user. These two approaches are discussed below.

A. Star Schema

The star schema is the simplest data warehouse schema, consisting of a single "fact table" with a compound primary key, with one segment for each "dimension" and with additional columns of additive, numeric facts. The name star schema is derived from the fact that the schema diagram is shaped like a star.The star schema makes multi-dimensional database (MDDB) functionality possible using a traditional relational database. Because relational databases are the most common data management system in organizations today, implementing multi-dimensional views of data using a relational database is very appealing. Even if a specific MDDB solution is used, its sources likely are relational databases.Another reason for using star schema is its ease of understanding. Fact tables in star schema are mostly in third normal form (3NF), but dimensional tables are in de-normalized second normal form (2NF). A start schema is represented in fig. 1 where Sales is the fact table. Account, Time, Product and Geography are the dimension tables and Dollar Amount and units are the measure.

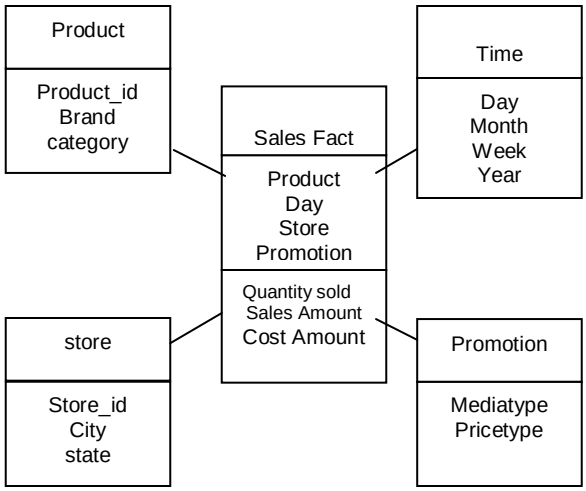

Fig. 1. Star Schema of Sales Data Mart

B. Cube

A data cube[10] is defined over a multidimensional space and consists of several dimensions representing the various perspectives of data. The data cube contains points or cells that are measures or values which represent a set of dimensions[9]. For example, consider a retail sales application where the dimensions of interest may include Product, Location and Time dimensions. A cell corresponds to the value for the corresponding Product, Location and Time. If the measure of interest in this application is sales amount, then a point represents the sales measure corresponding to the Product, Location and Time dimensions.

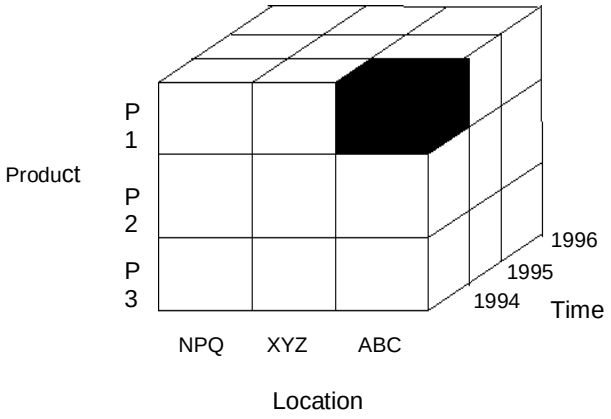

Fig. 2. Cube representation of Sales Data Mart

In Fig. 2 , the shaded cell corresponds to sales for Product `PI' in `ABC' for 1994.

III. DATA MARTS

The data mart has evolved from the data warehouse concept. It is a small data warehouse that satisfies the needs of a reduced set of users [4]. It is a specific, subject oriented, repository of data designed to answer specific questions for specific users. It is a collection of subject areas organized for decision support based on the needs of a given department. It only contains the specific data for local analysis. It is a simpler form of a data warehouse focused on a single subject (or functional area) such as sales, finance, marketing, HR etc. It represents data from single business process. There are many reasons for creating data mart .

TABLE I

REASONS FOR CREATING DATA MARTS

Reasons for creating Data Marts	
Ease of creation	• It is easy to create data marts as it is specific to a particular subject.
Response time	• Improves end-user response time.
Lower cost	• Lower cost than implementing a full Data warehouse.
Enhancement	• easy to enhance over time
Simplicity	• concentrate on a single subject.
Scope	• easy to understand, as scope is limited

Due to the above mentioned reasons data mart strategy can mitigate the risk, limit the expense, and reduce the time required to deliver data warehouse functionality. Data marts are typically application-specific databases designed to address particular question spaces, not to be data repositories this differs from most relational schemas, which generally act as source systems and are designed for redundancy and performance. There are various approaches for designing data marts.

IV. APPROACHES FOR DESIGNING DATA MARTS

There are mainly two approaches for designing data marts. These approaches are discussed below:

A. Dependent data marts

The first approach is to build dependent data marts (DDM). A dependent data mart is a logical subset or a physical subset of a larger data warehouse. According to this approach the data marts are treated as the subsets of a data warehouse [3]. In this approach, firstly a data warehouse is created from which further different data marts can be generated. These data marts are dependent on the data warehouse and extract the necessary data from it. In this approach as the data mart is created by data warehouse therefore there is no need of data mart integration. It is also known as top down approach. The fig. 3 represents DDM

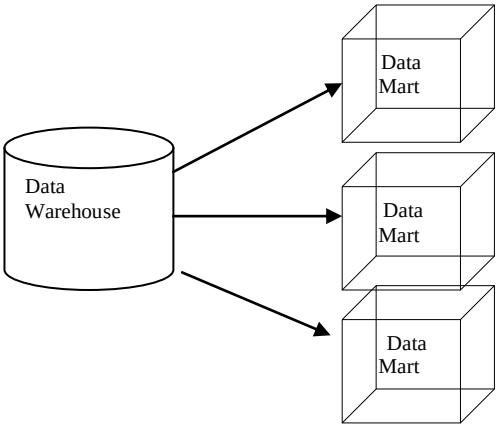

Fig. 3. Dependent Data Mart

B. Independent data marts

The second approach is independent data marts (IDM) Here, firstly independent data marts are created and then a data warehouse is designed using these independent multiple data marts [4]. In this approach as all the data marts are designed independently therefore integration of data marts is required. It is also termed as bottom up approach as the data marts are integrated to design a data warehouse. The fig.4 represent IDM.

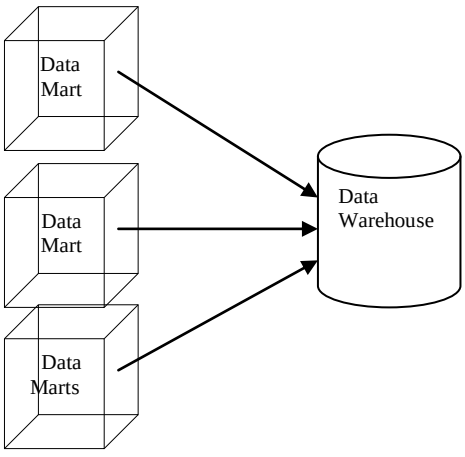

Fig. 4. Independent Data Mart

V. DATA INTEGRATION

Data integration is the process of combining data residing at different sources and providing the user with a unified view of this data.[5]. This process emerges in a variety of situations both commercial (when two similar companies need to merge their databases) and scientific (combining research results from different bioinformatics repositories). Data integration appears with increasing frequency as the volume and the need to share existing data explodes.There are different approaches for data integration.

A. Integration with Dimension Sharing.

A dimension of different data marts is shareable when the dimension is same in both the data marts. Having shareable dimensions and facts is important because it gives the ability to integrate the data marts and to combine and correlate such data. In particular, integration is based on joining multiple data marts over common dimensions [4]. Suppose we have a Sales data mart with dimensions Customer, Time and Store . Also let us have a second data mart named as Store Invention with dimensions Product and Store. To integrate these two data marts we choose a dimension that is common in both the data marts i.e. store, as the store is exactly the same dimension in both the data marts.

B. Integration with dimension compatibility

In this section, we discuss the impact of dimension compatibility on integration. According to this view two dimensions of different data marts are compatible when their common information is consistent. Similarly, two facts are compatible when their contents can be combined in a meaningful way. Having compatible dimensions and facts is important because it gives the ability to look consistently at data across data marts and to combine and correlate such data. [7][8]. Suppose we have a Sales data mart with dimensions Customer, Time and Store and the other data mart Warehouse Inventory data mart with dimensions Product and warehouse. Both the dimensions Store and Warehouse have common attribute say city. Before integrating them, they need to be aggregated over the common city level in the compatible dimensions Store and Warehouse**.**

C. Integration with generalization

As already discussed that to integrate data marts either the data marts share the exactly same dimension or they must be dimension compatible. According to above approaches we are unable to integrate the data marts if the dimensions are not same. However we argue that it is also possible to integrate data marts and drill across them if dimensions are not exactly the same [5] . If this is the case , semantic relationship should exit between the dimensions in the stars. Dimensions of different stars could be related by Generalization so that integration would be allowed. For instance Customer and Clerk are both specialization of People. Therefore we could travel from a star with Customer dimension to another one with Clerk dimension, if their sets of instances are not disjoint. This approach is suitable when the dimensions are not compatible. In the Fig 5 we can specialize People dimension at SaleRole level to get Clerk dimension which contains a level (i.e. clerk) with instances corresponding to people acting as clerks and another one with only one instance representing the set of all clerks.

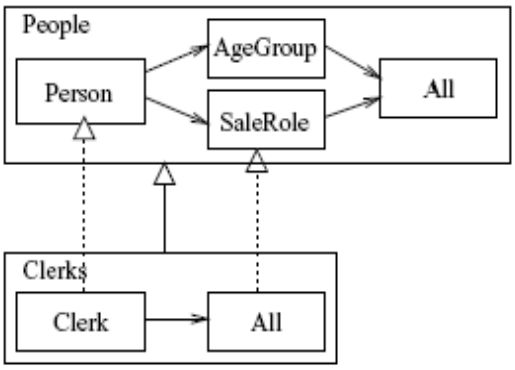

Fig. 5. Integration with generalization

VI. CONCLUSION

There are situations when data from different organization should merged for decision making purposes. That is only possible when two different data warehouse / data marts share a common dimension. We have shown the possibilities to combine different data marts. There are mainly two approaches of designing data marts one is dependent data mart and the other is independent data mart. The integration of data marts is required in case of independent data marts as they are created in different environment. There are other approaches of designing data marts like hybrid approach and federated approach. This paper discusses designing and integration of data marts. We emphasis on the design of data marts and requirement of integrating them.

REFERENCES

[1] W.H.Inmon Building the Data Warehouse. John Wiley and Sons, Secon Edition 1996.
[2] L.Cabibbo and R.Torlone. A logical approach to multidimensional databases,1998
[3] W.H. Inmon . Building the Data warehouse.John Wiley & Sons, Second Edition 1996.
[4] Ralph Kimball. The data warehouse tool kit , John wiley & sons, 1996.
[5] A.Abello, J. Samos, F. Saltor. On Relationships Offering New Drill Across Possibilities. In Int. Workshop on Data Warehousing and OLAP (DOLAP2002), ACM (2002).
[6] A.Abello, J. Samos, F. Saltor. Understanding facts in a multidimensional Object-Oriented Model. In Proc. of the 4th Int. Workshop on Data warehousing and OLAP (DOLAP), pages 32-39, ACM press, 2006.
[7] L.Cabibbo and R.Torlone. On the integration of autonomous data marts.16th Int .Conference on Scientific and Statistical Database Management (SSDBM'04), 2004.
[8] L.Cabibbo and R.Torlone .Dimension compatibility for data mart integration. SEBD, 2004, Pages 6-17.
[9] A.Datta and H.Thomas .The Cube Data model: A conceptual model and Algebra for on-line Analytical Processing in Data warehouses. In proceedings of the 7th workshop on Information Technologies and Systems (WITS'97), pages 91-100, 1997.
[10] J.C. Trujillo. The GOLD model: An Object Oriented Multi-dimensional data model for Multi-dimensional databases, 1999.

