Significance Of Data Warehousing

Mainak Dam*

Abstract

In this tinsel world of dazzling technology people are more and more bent on accuracy, efficiency and speed of the machines they use. In case of corporate sectors, business strategies should be chalked out lots of historical data in lesser amount of time. Obviously, we need a sophisticated store for our valuable data. Data warehouse is a technology that leads us to build an efficient store house for the data and improves the mode of business and analysis task. In the following sections some amazing aspects of this wonderful technology will be delved into.

1. What is a Data Warehouse ?

A data warehouse is an integrated database of analytical information culled from an organization's operation database. It is a collection of data in support of management decisions. A data warehouse organizes and stores a company's analytical information. It is the data that management studies and uses to analyze the business, to produce reports, to make forecasts and so on over a long historical time perspective. This analytical information is periodically extracted from the multiple operational databases. Data warehousing provides data access to the enterprise. It also tackles the problem of providing needed levels of information for the enterprise to survive and prosper in an increasingly competitive world.

2. A Solution, Not a Product

Often we think that a data warehouse is a product, or a group of products, that we can buy to help get answers to our questions and improve our decision-making capability. But it is not so simple. A data warehouse can help us get answers for better decision making, but it is only one part of a more global set of processes. For example, where did the data in the data warehouse come from? How did it get into the data warehouse? How is it maintained ? How is the data structured in the data warehouse? What is actually in the data warehouse? These are all questions that must be answered before a data warehouse can be built. Data warehousing is the design and implementation of processes, tools and facilities to manage and deliver timely, accurate, and understandable information for decision making. It includes all the activities that make it possible for an organization to create, manage and maintain a data warehouse or data mart.

3. Why Data Warehousing ?

The concept of data warehousing has evolved out of the need for easy access to a structured store of quality data that can be used for decision making. It is globally accepted that information is a very powerful asset that can provide significant benefits to any organization and a competitive advantage in the business world. Organizations have vast amount of data but have found it increasingly difficult to access it and make use of it. Because it is in many different formats, exists on many different platforms, and resides in many different files and database structures developed by different vendors. Thus organizations have had to write and maintain perhaps hundreds of programs that are used to extract, prepare, and consolidate data for use by many different applications for analysis and reporting. Also, decision makers often want to dig deeper into the data once initial findings are made. This

* *4th year student of Computer Science & Engineering*

would typically require modification of the extract programs or development of new ones. This process is costly, inefficient, and very time consuming. Data warehousing offers a better approach. Data warehousing implements the precess to access heterogeneous data sources; clean, filter, and transform the data; and store the data in a structure that is easy to access, understand, and use. The data is then used for query, reporting, and data analysis. As such, the access, use, technology and performance requirements are completely different from those in a transaction-oriented operational environment. The volume of data in data warehousing can be very high, particularly when considering the requirements for historical data analysis. Data analysis programs are often required to scan vast amount of that data, which could result in negative impact on operational applications that are more performance sensitive. Therefore, there is a requirement to separate the two environments to minimize conflicts and degradation of performance in the operational environment.

4. Short History

The origin of the concept of data warehousing can be traced back to the early 1980s, when relational database management systems emerged as commercial products. The foundation of the relational model with its simplicity, together with the query capabilities provided by the SQL language, supported the growing interest in what then was called end-user computing or decision support. To support end-user computing environments, data was extracted from the organization's online databases and stored in newly created database systems dedicated to supporting ad hoc end-user queries and reporting functions of all kinds. One of the prime concerns underlying the creation of these systems was the performance impact of end user computing on the operational data processing systems. This concern prompted the requirement to sepi rate end-user computing systems from transactional processing systems. In those early days of data warehousing, the extracts of operational data were usually snapshorts or subsets of the operational data. These snapshots were loaded in an end-user computing (or decision support) database system on a regular basis, perhaps once a week or once per month. Sometimes a limited number of versions of these snapshots were even acccumulated in the system while access was provided to end users equipped with query and reporting tools. Data modeling for these decision support database systems was not much of a concern. Data models for these decision support systems typically matched the data models of the operational systems because, after all, they were extracted snapshots anyhow. One of the frequently occuring remodeling issues then was to normalize the data to eliminate the nasty effects of design techniques that had been applied on the operational systems to maximize their performance, to eliminate code tables that were difficult to understand, along with other local clean-up activities. But by and large, the decision support data models were technical in nature and primarily concerned with providing data available in the operational application systems to the decision support environment. The role and purpose of data warehouses in the data processing industry have evolved considerably since those early days and are still evolving rapidly. Comparison of today's data warehouses with the early day's decision support databases should be done with great care. Data warehouses should no longer be identified with database systems that support end-user queries and reporting functions. They should no longer be conceived as snapshots of opeational data. Data warehouse databases should be considered as new sources of information, conceived for use by the whole organization or for smaller communities of users and data analysts within the organization. Simply reengineering source data models in the traditional way will no longer satisfy the requirements for data warehousing. Developing data warehouses requires a much

more thoughtfully applied set of modeling techniques and a much closer working relationship with the business side of the organization. Data warehouses should also be conceived of as sources of new information. This statement sounds controversial at first, because there is global agreement that data warehouses are read-only database systems. The point is that by accumulating and consolidating data from different sources, and by keeping this historical data in the warehouse, new information about the business, competitors, customers, suppliers, the behavior of the organization, business processes and so on, can be unveiled. The value of a data warehouse is no longer being able to do ad hoc query and reporting. The real value is realized when some one gets to work with the data in the warehouse and discovers things that make a difference for the organization, whatever the objective of the analytical work may be. To achieve such interesting results, simply reengineering the source data models will not do.

5. Characteristics of a Data Warehouse

According to Bill Inmon, author of Building the Data Warehouse and the guru who is widely considered to be the originator of the data warehousing concept, there are generally four characteristics that describe a data warehouse.

Subject-oriented : Data are organized according to subject instead of application e.g. an insurance company using a data warehouse would organize their data by customer, premium, and claim, instead of by different products (auto, life, etc.). The data organized by subject contain only the information necessary for decision support processing.

Integrated : When data resides in many separate applications in the operational environment, encoding of data is often inconsistent. For instance, in one application gender might be coded as "m" and "f", in another by 0 and 1. When data are moved from the operational environment into the data warehouse, they assume a consistent coding convention e.g. gender data is transformed to "m" and "f"

Time variant : The data warehouse contains a place for storing data that are five to ten years old, or older, to be used for comparisons, trends and forecasting. These data are not updated.

Non-volatile : Data are not updated or changed in any way once they enter the data warehouse, but are only loaded and accessed.

6. Processes in Data Warehousing

The first phase in data warehousing is to "insulate" current operational information, i.e. to preserve the security and integrity of mission-critical *OLTP* applications, while giving access to the broadest possible base of data. The resulting database or data warehouse may consume hundreds of gigabytes - or even terabytes - of disk space. What is required then are efficient techniques for storing and retrieving massive amount of information. Increasingly, large organisations have found that only parallel processing systems offer the sufficient bandwidth required for this.

The data warehouse thus retrieves data from a variety of heterogeneous operational databases. The data is then transformed and delivered to the data warehouse/store based on a selected model (or mapping definition). The data transformation and movement processes are executed whenever an update in the warehouse data is required. So there should be some form of automation to manage and execute these functions. The information that describes the model and definition of the source data elements is called "metadata". The metadata is the means by which the end-user finds and understands the data in the warehouse and is an important part of the warehouse.

It includes

i) structure of the data
ii) algorithm used for summarization and
iii) mapping from the operational environment to the data warehouse.

Data cleansing is an important aspect of creating an efficient data warehouse in that it is the removal of certain aspects of operational data, such as low-level transaction information, which slows down the query times. The cleansing stage has to be as dynamic as possible to accommodate all types of queries even those which may require low-level information. Data should be extracted from production sources at regular intervals and pooled centrally but the cleansing process has to remove duplication and reconcile differences among various styles of data collection.

Once the data has been cleaned it is then transferred to the data warehouse which typically is a large database on a high performance box either SMP, Symmetric Multi-Processing or MPP, Massively Parallel Processing. Number-crunching power is another important aspect of data warehousing because of the complexity involved in processing ad hoc queries and because of the vast quantities of data that the organization wants to use in the warehouse. A data warehouse can be used in different ways. For example, it can be used as a central store against which the queries are run or it can be used like a data mart. Data marts which are small warehouses can be established to provide subsets of the main store and to summarise information depending on the requirements of a specific group/department. The central store approach generally uses very simple data structures with very little assumptions about the relationship between data whereas marts often use multidimensional databases which can speed up query processing as they can have data structures which reflect the most likely questions.

7. The Data Warehouse Model

Data warehousing is the process of extracting

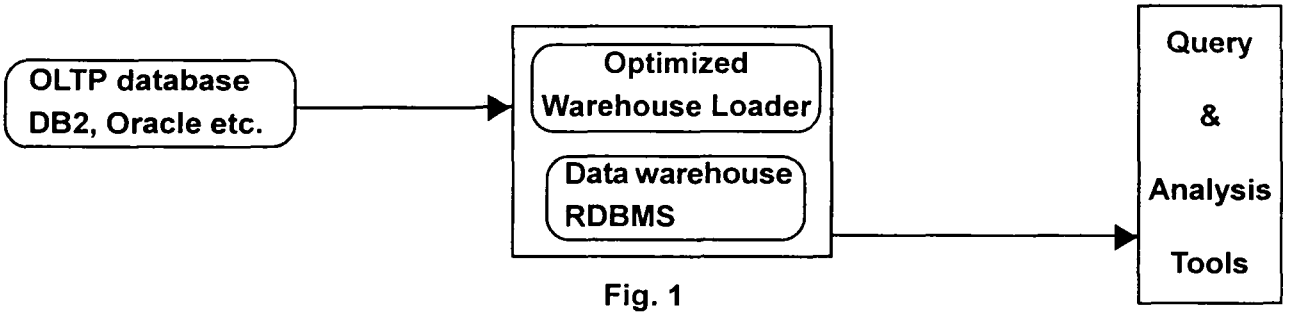

Fig. 1

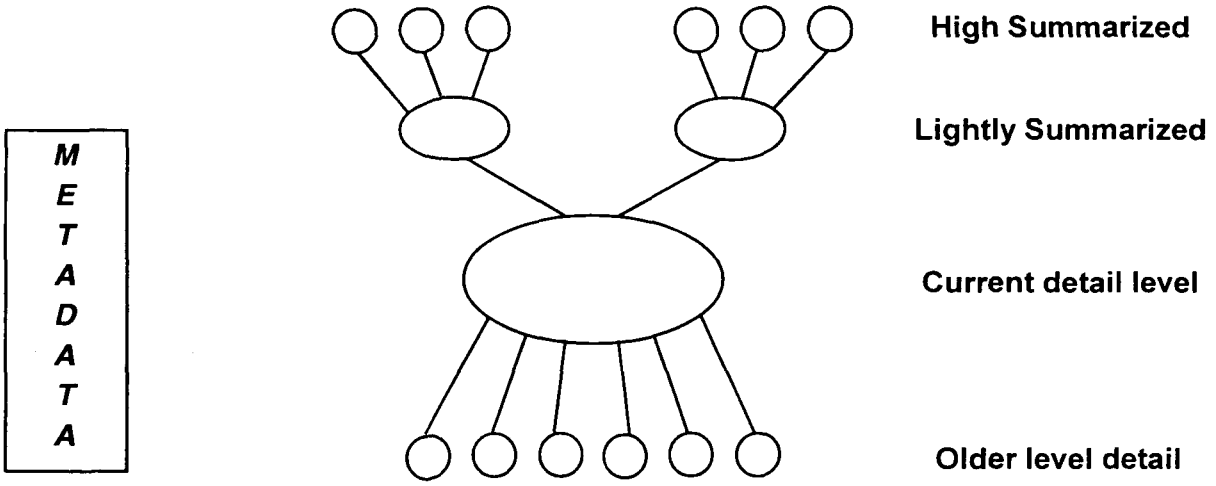

Fig. 2

and transforming operational data into informational data and loading it into a central data store or warehouse. Once the data is loaded it is accessible via desktop query and analysis tools by the decision makers. The data warehouse model is illustrated in Fig. 1.

The data within the actual warehouse itself has a distinct structure with the emphasis on different levels of summarization as shown in Fig. 2. The current detail data reflects the most recent happenings, which are usually the most interesting. Older detail data is stored on some form of mass storage, is infrequently accessed and stored at a level detail consistent with current detailed data. Lightly summarized data is data distilled from the low level of detail found at the current detail level and generally is stored on disk storage. When building the data warehouse we have to consider over what unit of time summarization is done and also the contents or attributes the summarized data will contain. Highly summarized data is compact and easily accessible and can even be found outside the warehouse. Metadata is the final component of the data warehouse and is really of a different dimension in that it is not the same as data drawn from the operational environment. This, on the other hand, is used as :

i) a directory to help the DSS analyst locate the contents of the data warehouse,

ii) a guide to the mapping of data as the data is transformed from the operational environment to the data warehouse environment,

iii) a guide to the algorithms used for summarization between the current detailed data and the lightly summarized data and the lightly summarized data and the highly summarized data, etc.

8. Schemas of Data Warehouse

To build the data warehouse the most widely used schema is Star Schema. It contains two types of tables, one major or fact table and many minor or dimension tables. Fact table holds all the primary keys of dimension tables and a major field which provides the summary information about a given subject. Thus fact table can be thought of as a primary key generator table. All the dimension tables are radially joined to the fact table.

Let us consider an example of star schema. Supose *'sales'* is a fact table and *'customer' 'product'*, *'time'* and *'store'* are dimensions. Relationship between fact and dimension tables is one-to-many. Fact table contains quantitative, measurable data; for e.g. amount of product sold. The fact table contains the *atomic* data; it may as well contain partially consolidated data.

The fact table keys are generated which allow flexibility and high performance over composite or concatenated keys. The generated key is numeric and acts as primary key of the fact table.

Dimension tables are smaller and contain more qualitive data like customer, product etc. The non-key fields of dimension tables are called attributes. It is possible that dimension tables will have their own dimension tables. In this case, the store dimension will contain district ids and region ids which will be referencing district and region dimensions of store dimension respectively. This schema is called **Fact Constellation** schema.

Advantages of star schema are

i) It is easy to define

ii) It reduces number of physical joins

iii) Provides very simple metadata

Drawbacks of star schema are

i) Summary data in fact tables (such as sales amount by region or district-wise or year-wise) yields poor performance for summary levels and huge dimension tables.

ii) Dimension tables must carry a level indicator for each record (unless they have sub-dimension tables) whenever aggregates are stored with detail facts. For example, in a store dimension, level helps identify whether product sale is region-wise or store wise.

Without level column indicator, keys for all cities in mid-west region including aggregates for region and districts will be pulled from sales table giving wrong results.

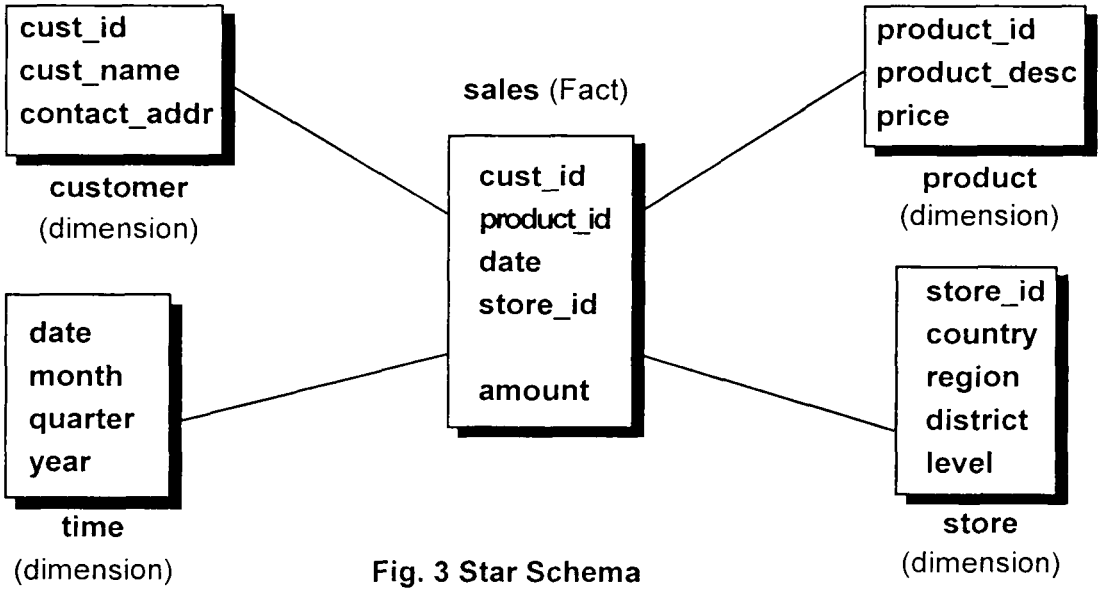

Fig. 3 Star Schema

Column called "level" is needed in all dimensions when aggregates are stored with detail fact tables. For example, to get sales amount by district, following query will be used.

SELECT a store_is, a. year, a. amount FROM sales a
WHERE a. store_id IN (SELECT b. store-id FROM store b
WHERE b. region = "MidWest" AND b. level = 2

For example, to get sales amount by store, following query will be used.

SELECT a store_id a. year. a. amount FROM sales a
WHERE a. store_id IN (SELECT b. store_id FROM store b
WHERE b. region = "MidWest" AND b. level = 3)

The **Snowflake** schema has normalised dimension tables by attribute level with each smaller dimension table pointing to an appropriate aggregated fact table. Each dimension table has one key for each level of dimension's hierarchy. For example, the lowest level key, the store key of store dimension will join store fact table which will also have product and time keys. the aggregation will be as follows :

Store → District → Region → Country

Practical way of designing a snowflake schema is to start with star schema and create snowflakes with queries. So separate extracts are not needed and referential integrity is inherited from dimension table. The SQL query given above can avoid level indicator if it uses snowflake schema. In snowflake schema, stores dimension will have store fact table, district fact table and region fact table. So the required amount can be directly extracted from sales store and district tables.

Advantages of Snow flake schema is that it provides best performance when queries involve aggregation.

Disadvantages of Snowflake schema are

i) Maintenance is complicated

ii) Increase in number of tables

iii) More joins will be needed.

Galaxy schema is a collection of many star schemas. Galaxy is used when informational structure does not fit into a single fact table and more fact tables are needed.

9. Conclusion

The need of an efficient data warehouse is increasing by leaps and bounds. Different companies with different business interest started incorporating a data warehouse in their organisation. Leading software vendors like Microsoft, IBM, ORACLE and many others have already introduced standard data warehouse solution in the market. The success rate is quite satisfactory and it is not far when having a standard data warehouse in a company will become a regular fashion.

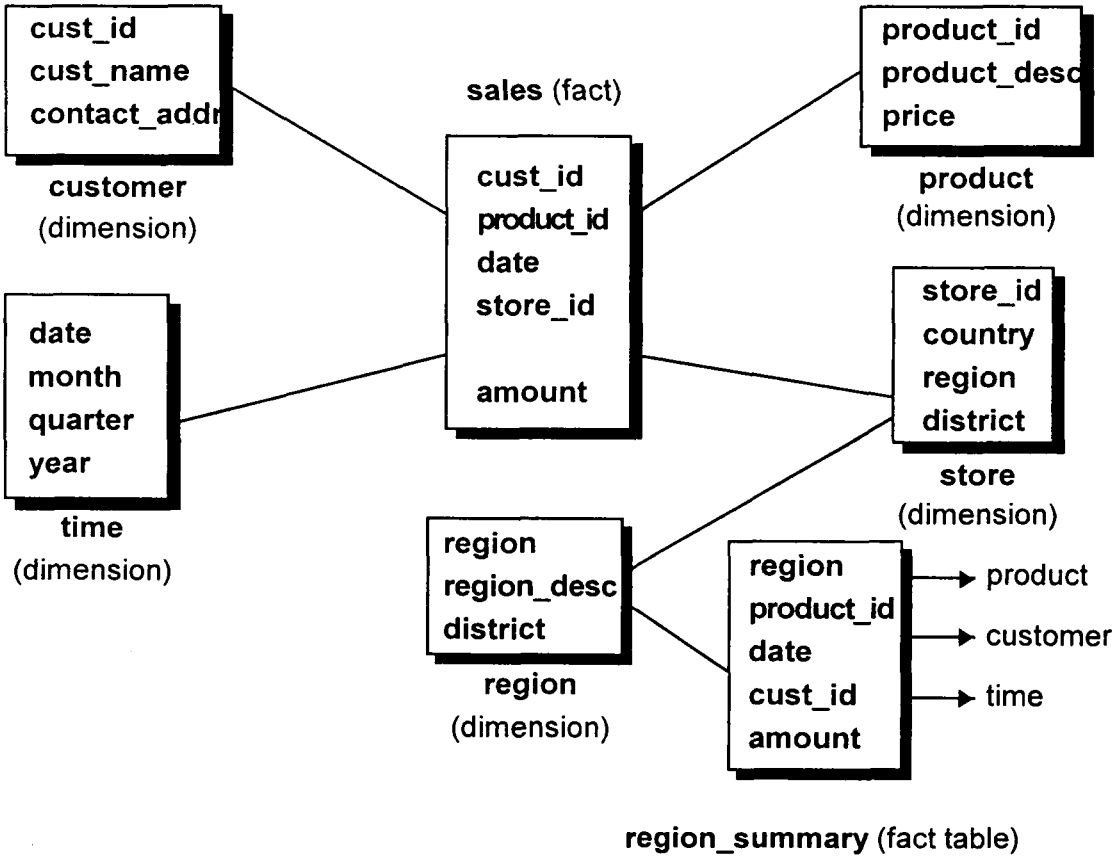

Fig. 4. Snowflake schema

'Knowledge has to be improved, challenged and increased constantly, or it vanishes.'

- Peter F. Drucker