A METHODOLOGY FOR DATABASE-ORIENTED DECISION SUPPORT SYSTEMS

Chowdhury Shamsul
WEH College of Business Administration
Roosevelt University, Albert A. Robin Campus,
1400 North Roosevelt Blvd., Schaumburg, IL 60173, USA
E-mail: schowdhu@roosevelt.edu

Ismail Waleid
Computer Networks Systems, ITT Technical Institute
7040 High Grove Boulevard, Burr Ridge, IL 60521, USA.
Email: wismail@mail.roosevelt.edu

ABSTRACT

Database applications, data warehouses, and data mining tools have quickly evolved into a unique and popular business application class. Decision makers already consider these systems to be key components of their IT strategy and architecture. The paper presents a methodology for implementing database systems in a decision support environment. E-R modeling is the obvious choice for implementing database systems. However, center-out or dimensional modeling schemas are the means that should be adopted for implementing database-oriented system for decision support, for example; in a data warehouse environment with very large database contents. Center-out modeling is also recommended for implementing small-scale decision support database systems in SME's.

Keywords: Database applications, DSS, Center-out and E-R modeling.

INTRODUCTION

A database application is not a single product or a solution. Rather it is a compilation of methodologies and technologies that together can provide a pragmatic and efficient approach to dealing with the problem of user-friendliness and accessibility to information that has been disseminated across the organization. As organizational requirements grow and change, database applications must evolve with them. Delivering on the promise of database application is not swift, low-priced, or simple. It requires dedication and core business knowledge (8).

Data modeling is the tool utilized to create a logical description of the data in a database. Most data modeling tools today support the Entity-Relationship (E-R) modeling technique that seeks to group data together in common relationships with the primary focus of eliminating redundancy in the database using the normalization technique (4,10,15). E-R modeling is very efficient in the Online Transaction Processing (OLTP) environment where the goal is to get the data into the database in the most efficient way possible. E-R diagrams identify and model business processes but are very transaction oriented and not at all subject oriented. They are also overly complex environments to query for ad-hoc user reports. E-R modeling techniques and rules of normalization will not give end users these requirements in a decision support system. In a decision support environment a data warehouse and/or a data mart are most commonly used. The data warehouse/data mart data are being extracted from existing OLTP systems within the organization. Data in a database contain both operative and directive information where as a data warehouse/data mart contain only directive information for decision making purposes and data are (5,12,13):

- time dependent
- non-volatile (never updated only query)
- subject oriented and
- integrated

Decision-making data are extracted from different OLTPs/external sources and further organized as per fact/s or burning question/s for decision-making purposes.

DATA WAREHOUSES AND DATA MARTS

Data warehouses are read-only, integrated databases designed to answer comparative and "what if" questions. Unlike operational databases that are set up to handle transactions and that are kept up to date as of the last transaction, data warehouses are analytical, subject-oriented and are structured to aggregate transactions as a snapshot in time.

Data marts (these are data warehouse subsets in their own right) are often built to meet particular needs. We may think of it as a miniature data warehouse designed for a particular area. The data itself may come from an enterprise wide database or a data warehouse. For example, the Sales department might only care about customer, product, and sales data, and so forth wants a certain data mart or data warehouse subset. Within the customer portion they might specifically care about identifying areas of customer satisfaction and frustration that could form another data subset and this could be carried out forming a chain of subsets depending on the size of the organization.

CENTER-OUT MODELING

In developing an E-R model for a database two general strategies are usually being used (2,6,10):

1. Top-down development: General requirements to specific requirements and provides a global /enterprise wide perspective.
2. Bottom-up development: Specific requirements to general requirements. It is typically faster and less risky

However, E-R modeling using either top-down or bottom-up strategies is not appropriate for modeling decision support databases, for example a data warehouse. We need a center-out strategy for modeling a decision support database (6). The center-out strategy focuses on the main burning points or questions that the model has to provide answers. This strategy matches the dimensional modeling or star schema as shown in figure 1.

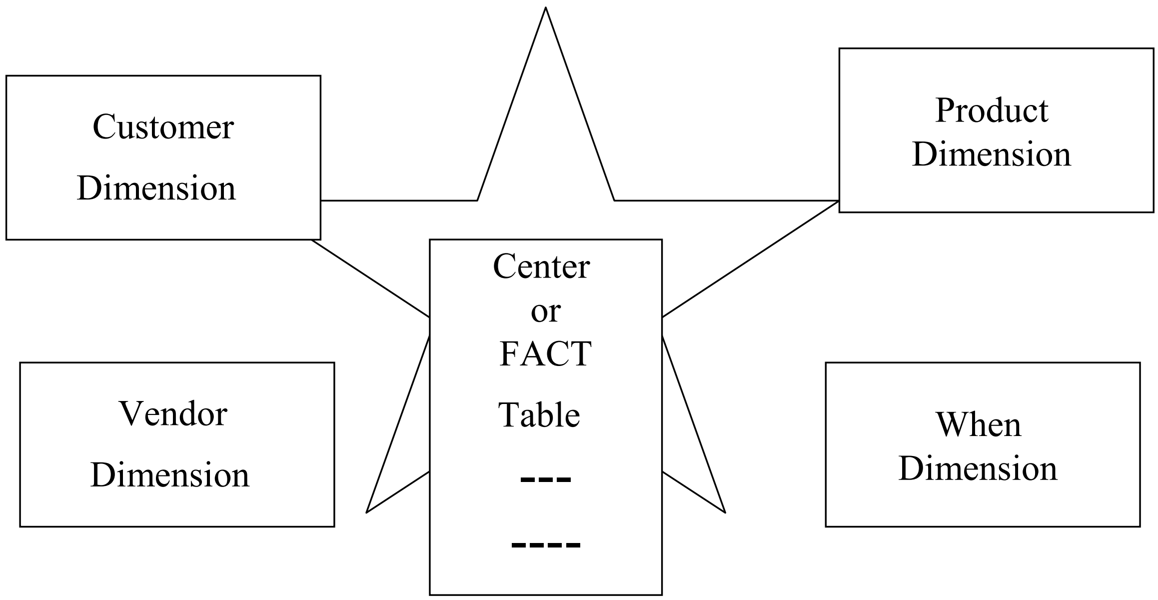

Figure 1: Center-out strategy for developing a DSS (1)

As data warehousing is a developing science with its own set of rules and procedures, "center-out modeling" and "star schema" may replace E-R diagramming as the most effective choice for warehouse design.

Star schemas have many advantages over the traditional normalized ER schema design. We believe the easiest and simplest way of understanding our target data is by viewing our whole data as amassed numbers broken down and classified by criteria. Backing up our intention, we propose to initially build our warehouse in the very same format. That's exactly what we do with the star schema. It is important to realize that OLTP is not meant to be the basis of a decision support system. It is all based on transactions, and a transaction is not about performing complex analysis to spot trends, however, it is about maintaining order lists, and assist in the inventory sale process. Hence, rather than exhausting our OLTP system by performing massive, expensive queries, we build a database structure that maps to our intention of viewing data (2).

First and foremost is performance. Let's assume that we have a *product, time and store* dimensions. If we assume we have ten years of daily data, 200 stores, and we sell 500 products, we have a potential of 365,000,000 records (365 X10 days X 200 stores X 500 products). As you can see, this makes the fact table long (14). The dimension tables and the fact table are heavily indexed on their key structure and these indexes drive all queries of the fact table. By selecting facts from the fact table and driving the query into the fact table with one or more dimension table keys, no more than one highly efficient index driven pass of the database is needed to satisfy a users query constraints. A query through millions of records can be satisfied fairly quickly in this environment. Most ad hoc query tools, including Oracle Discoverer, are designed to perform optimally in this star schema environment and don't necessarily perform well within traditional OLTP schemas (8,9).

Should the need for growth arise, fact table and dimension tables can easily be modified to accommodate additional dimensions and records.

MODEL DEVELOPMENT METHODS

Three methods are in use for dimensional modeling (13):

1. Development by modification: Buy a home that is constructed and knock some walls to add new walls, etc.
2. 2. Development from template: Buy some existing home plans and have a house built to the plans or a modified version of the plans.
3. Full custom development: Contact an architect and start from scratch.

However, development by modification is in common use and the success depends on factors, like:

- the models are new, well designed and normalized
- the IT staff is very familiar with the source system OLTP models
- the IT staff understands the data that is already in the database
- many DSS (Decision Support System) database (warehouse) reference and fact tables are already in the OLTP model in a different form

MODEL IMPLEMENTATION

In a project work conducted at our Business College at Roosevelt University (6,11) the modification by development approach has been used to develop a star model for decision support by modifying the existing E-R model of the NorthWind OLTP database of a small-scale enterprise. The purpose of the star model was to provide answer to those burning business questions which were rather difficult, if not impossible; and time consuming to obtain from the OLTP database based on E-R modeling. The conversion from E-R to Star was successful because the factors that have been mentioned to above were considered carefully and the project's team members were familiar with the E-R model and the data in the OLTP database.

Star or dimensional modeling is mainly being adopted for implementing database-oriented system for decision support, for example; in a data warehouse environment with very large database contents. Dimensional modeling is also recommended for implementing small-scale decision support database in SME's (Small and Medium Enterprise).

SHARING A SINGLE STAR AMONG MULTIPLE DSS

Can various businesses share their "STARS" together? Or each organization needs to "STAR" its' own future? And if either was the case, in what form(s) or level(s) can the sharing occur?

The father of the Data warehouse, Bill Inmon (7) has artfully answered these questions in the following words: "The advocates of dimensional modeling would like to believe that a single star join can be created for the optimization of everyone's requirements. Such is not the case at all" furthermore, he proceeded saying “Because the star join is shaped around user requirements and because user requirements vary from one type of user to the next, not surprisingly different star joins are optimal for different types of users”.

Factors such as: Data Sequence/Definition, Data Demography, Technology, and Time Dimensions may profoundly shape up the form of the Star in demand per a given organization.

SUMMARY OF A CURRENT INDUSTRY CASE ANALYSIS

A data mart is very much like a data warehouse but is usually smaller in size and focused on a single purpose, such as sales, operation, or promotion and advertising. Because of its reduced size, Data marts may earn back their cost in a shorter time period than enterprise-scale data warehouses may.

The aforementioned reasons were sufficient to persuade our firm (XYZ Inc.) to launch a series of Data Marts. XYZ is a major healthcare consulting chain with outlets organized into regions. Consequently, they set us their Data Marts with a region-to-outlet hierarchy dimension.

XYZ wants to be able to adopt the Top-Bottom view—sales by region—into more detailed levels, such as sales by outlet. Utilizing their existing data marts, regional managers can query the sales of a given region and realize its sales volume and whether sales are off in general in a particular region or if a particular outlet is not doing well. Each outlet belongs to a particular region at any given time that is reflected in the outlet manager's report to regional managers.

But XYZ has a typical problem: regions are frequently reorganized. As a result of, their old Data Marts are currently unable to answer current queries. Data Marts are built to support "specific" questions or queries. They are typically not duplicated. Every time a new type of user (typically manager or customer) question arises, a new transformation of the data mart will arise. Every time a new type of data arises, a new alteration will arise. The final result is a complex series of unrelated data marts, each with a large amount of infrastructure investments to handle the additional indexing structures. XYZ continued to predict questions and build Data Marts to answer their predefined questions. That continued until they faced a question that no Data Mart can ever answer and it was: Are our Data Marts making us money or costing us revenue?

Several prepositions were put on the table and it seems the winning one was preaching for building a huge Data Warehouse to consolidate all warehouses and their associated subsets.

We think NOT! It is obvious that by adopting this solution, they would just be reinventing the wheel and duplicating their own initial mistakes. Along with the short-term gains, we must also focus on the long-term pains (3).
In order to embrace on this never-ending cycle of change, the Data Warehouse needs to quickly react to new conditions and challenges. In the data mart environment this reaction time is severely impacted. An environment where users can ask questions directly against business modeled detailed data will allow for maximum advantage in a competitive arena.

DISCUSSIONS: ISSUES TO THINK ABOUT!

The goal behind a true data warehouse is to provide the end user with the ability to ask any question against any business data in order to make decisions that can not currently be made today: Can this goal be reached with all outcomes satisfied?

How are we calculating cost? What are our parameters?

Data representation issues in businesses and what is the most logical representation of data in various businesses?

Data integrity issues i.e. Data which may not have been 'cleaned' aptly may lead to flawed conclusions.

The issue of statistical inference - can we correct statistical inferences based on archived data? How do we handle noisy and incomplete data?

What can be understood from the patterns that emerge? How do we represent this information to users? How will humans interact with this information?

Some organizations have started building the data marts first, so that they can benefit much sooner from data warehousing. In that case it would be necessary that they would have to adopt the “bottom up” approach to building their enterprise warehouse as opposed to a “top down” approach. However there is a potential problem that the data marts might be difficult to integrate.

REFERENCES

1. Agosta, L. (2000). The Essential Guide to Data Warehousing. Prentice Hall PTR, Upper Saddle River, NJ 07458.
2. Anahory, S and Murray, D. (1997). Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems. Addison Wesley Longman Limited, England.
3. Armstrong, R. (2002). White paper by NCR Corporation - The Fallacy of Data Mart Centric Strategies (Short Term Gain, Long Term Pain).
4. Chen, P. (1976). The Entity-Relationship Model – Towards a Unified View of Data, ACM Transactions on Database Systems, pp 9-36.
5. Chowdhury, S (1990). Computer-Based Support for Knowledge Extraction from Clinical Databases, Linköpings Studies in Science & Technology. Dissertations No. 240.
6. Chowdhury, S (2002). Lecture notes on Data Warehouse and Data Mining. The College of Business Administration, Roosevelt University, IL.
7. Inmon, B. (1999). “The Problem with Dimensional Modeling” DM Review Magazine Archived Article.
8. Ismail, W. and Chowdhury, S. (2003). Database Applications in Business. In Proceedings of the MBAA, Chicago, IL.

9. Kimball, R. (1996). The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses, John Wiley.
10. Kroenke, D. (2002). Database Processing – Fundamentals, Design and Implementation (8th ed.). Pearson Education, Inc.
11. Letowski, B. Parzatka, H. Woods, N (2002). NorthWind Star Schema – a project work presented and submitted in Seminar on Data Warehouse and Data Mining at the College of Business Administration, Roosevelt University, IL. 2002.
12. McFadden, F. Hoffer, J. and Prescott, M (1999). Modern Database Management. (5th ed.). Addison-Wesley Educational Publishers, Inc.
13. Sperley, E (1999). "Planning, Building, and Implementation". The Enterprise Data Warehouse. Prentice Hall.
14. Utley, C. (2002). “Designing the Star Schema Database” Data Warehousing Resources, November 2002.
15. Vandivier,S. (2001). President, AVANCO International, Inc. http://www.avanco.com/d_warehouse.htm#Steve, 2001.