Relational versus multidimensional databases as a foundation for online analytical processing

Lessons from two case studies

David Dodds, Helen Hasan & Edward Gould
Edward_Gould@uow.edu.au
Department of Business Systems
University of Wollongong

Abstract

The huge volume of data stored in organisational databases is no longer seen as a data management problem, but rather as a potential company asset to be exploited for information. As a result there is renewed interest by IT practitioners in data models and database structures. Multi-dimensional forms in particular have joined their relational counterparts as legitimate tools for extracting vital business information from company data. This paper compares the conceptual differences between two common methods used for exploiting company data, namely multi-dimensional on-line analytic processing (MOLAP) and relational on-line analytic processing (ROLAP).

Keywords: data warehouse, multidimensional database, OLAP, ROLAP

BRT Keywords: EA, CB06, HA03

Introduction

The familiar and well tried relational database model is lately undergoing a process of upheaval in the Decision Support Systems (DSS) market due to sweeping changes in the requirements of users. This is due, in part, to the fact that the traditional row and column structure of relational databases is two dimensional requiring complex joins to link data from different tables. In order to perform sophisticated analysis of data in these systems the services of professional programmers are usually required to provide customised views of specific segments of the database. Developments aimed at overcoming this problem, and at making the analyst feel closer to the data, have taken place in the areas of data warehousing, multi-dimensional databases and on-line analytic processing (OLAP).

As David Baum (1996) points out, the theory behind a data warehouse is to separate the day to day activities of production applications from the operations carried out by knowledge workers for reporting and analysis. Warehouses are organised by subject rather that application and hence, provide better support for the process of decision making. The data for these warehouses needs to be downloaded from operational systems usually during off hours on a periodic basis and is performed either as a bulk download or a change-based replication of the differences between what exists

in the warehouse and what has taken place in the operational system during the previous time period. The processing speed and capacity of current hardware now enable database software to run efficiently and effectively so that hardware capability is not the focus of concern (Kimball, 1994). The data most commonly stored in an MDDB is an organisation's historical performance figures for use in management decision making. Data warehouses tend to have multifaceted architectures, the simplest of which are designed on two or three levels. Separate layers handle the warehouse database and other layers control a multidimensional OLAP engine and client interface for decision support analysis either combined in the case of two level architecture or separate in the case of three.

As is often the case with the introduction of new technology what is happening in practice is leading research into the phenomenon. Nowhere is this more evident than in the area of multi-dimensional databases (MDDB). Report published in popular IT magazines or as white papers by specific MDDB product vendors cover mechanisms and techniques for efficient on-line analytic processing (OLAP) (for example Creeth & Pendse, 1995) or stories of successful implementation of MDDB. Many of these are aimed at convincing executives and business analysts of the merits of the approach (for example Armstrong, 1990, Gentia, 1998). However, there is a need for a more objective and theoretical analysis of this area, particularly on aspects that concern application developers and end-users rather than creators of multi-dimensional systems.

This paper will present two case studies involving the development of MDDB applications each using a different methodology, one top-down and the other bottom-up. The comparison of these methodologies raises a number of issues for discussion and further study. The paper aims to shed light on the manner in which an organisation's data and information can best be used to support the progress of the organisation towards attaining its business objectives.

MDDB and OLAP: Descriptions and Definitions.

There is some confusion in the popular use of a number of terms associated with MDDB and OLAP. This section will describe these terms as they are used in this paper and are based on a combination of definitions from the literature and that experienced by the authors in common practice.

In MDDB, data is stored in such a way as to be represented to the user as a hypercube or multi-dimensional array, where each core data value or fact occupies a cell indexed by a unique set of dimension values. In its simplest form this can be visualised using a fact (such as number of products sold) along the three most common dimensions (time, location, product type). This representation can be extended to include any number of facts and dimensions and is in contrast to the set of tables used to represent data in the well-known relational database (RDB) model. The contrast between RDB and MDDB will be elaborated in the following section.

The resulting hypercube of information in a MDDB can be viewed and manipulated with the help of an interactive graphical user interface (GUI). Techniques such as *slicing* and *dicing* as well as *rolling up* or *drilling down* the dimensions give different views of the data. Because of this interactivity the term OLAP is often used widely and loosely among database practitioners and in the popular literature interchangeably with that of MDDB. Although the terms are interconnected, OLAP systems gather data for making decisions about the long term workings of an organisation

and hence provide information support for managerial decision making such as customer profiling, forecasting and trend analysis. However, there is disagreement in practice concerning the exact characteristics essential for an application to be called OLAP.

In conjunction with Arbor Software, Ed Codd proposed a set of features that defined OLAP (Codd, et. al., 1993). However, despite Codd's reputation in the relational database field, his association with a particular OLAP vendor meant that neither researchers nor practitioners gave much credence to this definition. A more recent definition from Creeth and Pendse (1995) called Fast Analysis of Shared Multi-dimensional Information (FASMI) and not dependent on a particular technology or application is now gaining acceptance. Even though this definition is based on a multi-dimensional representation of data, the debate between the roles and merits of the relational and multi-dimensional data models continues. This has led to the concepts of ROLAP (relational OLAP) and MOLAP (multi-dimensional OLAP). Both ROLAP and MOLAP are legitimate ways of representing data to the user in a multi-dimensional form and afford a logical consolidated data-set with a GUI user interface

These issues will be discussed as a prelude to the case studies, one of which involves the translation of data from a relational to a multi-dimensional database for the purpose of information analysis.

Multi-dimensional and relational data models and methodologies

It is our contention that RDB and MDDB are complementary not competing database architectures. MDDB has been described in the previous section of the paper and, as the relational data model has been studied and used for many years, it is assumed that the reader is familiar with its table structure using the concepts of normalisation and entity-relationship (ER) analysis. (see also Kimball, 1997 for a comparison of dimension and ER modelling)

RDB management systems (RDBMS) have been adopted by most organisations for their OLTP. By normalising the data, redundancy is eliminated and business transaction records captured in very little time. In most medium and large organisations such databases typically grow to an enormous size and contain data that have the potential to tell management much about the state of the enterprise. Information can be extracted from the database via reports generated through the RDBMS or retrieved ad hoc by means of the structured query language (SQL). However due to the fragmentation of normalised data an incredible number of joins are needed to satisfy even moderately complex queries that a company executive might ask, a phenomenon known as "runaway SQL" (Lazer, 1996).

Decision Support Systems (DSS), date warehousing and EIS were all computer applications introduced in the hope that they would enable executives or company analysts to more easily retrieve meaningful business information that exist inside disparate enterprise databases. The concept of a unified logical corporate data model was popular in the 1980s (Brancheau & Wetherbe, 1986) and the 1990s saw many of these systems physically implemented as a data warehouse (Shanks et al, 1997). Among those experienced in these technologies it is commonly felt that that the idea of dimensions is a logical way to view data at the corporate level (Frank 1994) and database software products have been available over the last few years based on the multi-dimensional structure.

The popularity of these MDDB products in the 1990s has raised two points of debate. The first of these is whether MDDB should or can be a replacement for RDBMS.

On this point it is now widely accepted that RDBMS are still necessary for OLTP and that, in most circumstances companies, should not replace their existing RDBMS with an MDDB product.

The second point of debate is more contentious, that of the relative merits of MOLAP and ROLAP as the preferred OLAP tool to perform business analysis with data sourced from the relational OLTP system. The fundamental point of distinction here is between data storage and processing capability. MOLAP stores a processed copy of data uploaded from source organisational databases to populate its own data structure. This has the advantage that the data can be cleansed and multiple aggregations performed during the upload so that optimum performance and flexibility is achieved for the user. ROLAP on the other hand analyses the original data in the current organisational database or a relational data warehouse so that the user can theoretically drill down to the unit data level usually by means of SQL extensions. However, processing power to do such analysis on the fly is enormous, resulting in a need for expensive high performance hardware. There can be a lack of historical data in ROLAP systems that use only the current organisational databases. ROLAP does have the advantage of being tied to the open systems standards of the underlying RDBMS and one common criticism levelled at MOLAP is the lack of standardisation among the proprietary MDDB products. However this drawback is minimised by the fact that most MOLAP databases are read-only and have the capability to automate the process of pulling data from all standard RDBMS. Other more pervasive problems with MOLAP are highlighted in the case studies which follow.

Comparing ROLAP and MOLAP : Two Case Studies

In this paper we would like to draw out the conceptual different between the common methods used for the development of MOLAP and ROLAP databases. MOLAP encourages a top-down approach first focusing on business problems, then identifying performance measures and dimensions of interest to business executives and analysts. A multi-dimensional meta-data model is then built often before sources of the relevant data are found. A number of authors have developed detailed methodologies for EIS development based on this approach and these were used in the first case study described below. ROLAP on the other hand, is a function sitting on top of a RDBMS and encourages a bottom-up analysis to identifies candidate facts and dimensions in the existing relational data models of the operational databases. This approach suits the traditional database designer familiar with the relational model and is essentially the same as an EIS but using an automated tool to convert data from a source RDB into a MDDB as will be described in the second case study. It should be noted that although the top-down or bottom-up approach can both be used for either MOLAP or ROLAP, there is definitely a preference for the top-down business focus with MOLAP and the converse with ROLAP.

Each case study concerned the development of a prototype EIS related to a specific business problem in an organisation. The developments were carried out as one year research projects by two part-time graduate MBA students, supervised by an experienced EIS researcher (see Hasan & Gould, 1994, Hasan & Lampitsi, 1995, Hasan & Hasan, 1997). Both student-developers were experienced IT professionals and worked in the organisations of their respective case studies so that the context of the problem was familiar to them. Each development was based on a methodology derived from the

literature read by the student and was implemented in an advanced object-oriented (OO) commercial EIS development tool.

Descriptions of the Case Studies

The first case (MOLAP) is the development of a system to analyse categories of problems dealt with by the help desk of an IT organisations in a large multinational company. The organisation was at the time using a flat file-based problem management system to register, track and monitor problems. One of the support managers, familiar with the short-comings of the existing system, was willing to act as executive sponsor of the new prototype. This manager was interviewed several times by the student-developer to gather requirements and to evaluate the final prototype. The original multi-dimensional structure used by the commercial package was used to implement this prototype.

The second case (ROLAP) is the development of a system to monitor and analyse student data in a large university. The data included demographic information as well as student course records and was stored in a commercial relational data-warehouse package whose design was based on government reporting requirements. There had been an attempt to create a user-friendly front-end to this database but the resulting system was used only by a few trained administrators. Senior management queries continued to be handled by the IT staff in an ad hoc fashion. A copy of the database, with student identities disguised, was provided for the development of the prototype by the database manager who acted as operating sponsor for the project. The prototype was implemented in a commercial MDDB which comes with a tool to transform source data from the RDB into the MDDB.

Methodologies for the Case Studies

In the MOLAP case study the focus was on the business problem and a top-down development methodology was employed based on the standard EIS literature including the work of Rockart and De Long (1988), Burkan (1991), Volonino and Watson(1992) and Barrow (1992). They describe EIS development as starting with the identification of an executive champion or sponsor and then determining the Critical Success Factors (CSF) and Key Performance Indicators (KPI), either of individuals or of the corporation, to establish the initial requirements of the EIS. A multi-disciplinary EIS team is then set up and an evolutionary prototyping method is used to continue the development process.

Since only a realistic subset of data is required for the initial prototype in an evolutionary development process, not much attention was paid to the availability and quality of the source data for the eventual EIS. The student-developer of this case study had access to the business manager who acted as project sponsor and who provided requirements, from which the facts and dimensions were determined, as well as realistic sample data. A simple MDDB model was adequate to set up the required facts and dimensions and the sample data could be loaded into the commercial DB from a comma delimited text file by a small program that was soon mastered by the student-developer. The object oriented approach then made it easy to set up a GUI interface to the DB with standard EIS capabilities as described above.

In the ROLAP case study the data was larger and inherently more complex, already structured in relational tables. Because of the involvement of the database

manager familiar with the data and the availability of the translation tool, the student-developer chose a bottom-up approach based on a multi-dimensional modelling framework prevalent in the literature (Kimball, 1997, Pokorny, 1998). This begins with an analysis of data in the existing relational database to identify facts and dimensions that relate to a subject of interest.

The key concept of the framework are two kinds of tables: the Fact Table which consists of the numerical measurements that exist within the database and Dimension Tables which are more descriptive data items that map to the natural dimensions within the business. The Fact Table is made up of multi-part keys that link back to the Dimension Tables giving a layout referred to as a star or snowflake schema. To successfully translate data from a RDB into a meaningful MDDB it is necessary to identify star schema within the RDB related to subjects of interest to the business analysts. The translation process is essentially one of de-normalisation, and hence simplification, and should in principle be capable of automation.

A development method suggested by Shanks and O'Donnell (1998) was used to implement this case study. Two subjects were identified: course enrolment and subject enrolments. The marketing manager was interview to determine some basic information requirements. However it was the implementation step, using the commercial DB tool that constituted the major time and effort of the project due to missing or ambiguous data in the relational database from the multi-dimensional perspective.

Common Features of the Case Studies

Developers and users of MDDB anticipate that the dimensional view of organisational data will provide managers with a better means of understanding the current state and future possibilities of their business. It was interesting that all the IT professionals involved in the case studies reported that they had difficulties with the basic concepts of facts and dimensions as well as identifying them in their data. On the other hand the student-developer in the MOLAP case study reported that he was surprised how quickly his executive sponsor grasped the dimension concepts in relation to his data. This suggests that familiarity with the data and the business problem is more significant than traditional database expertise in top-down multi-dimensional modelling.

Once facts and dimensions were identified in the MOLAP case study it was relatively straight forward to set up the data model (or meta-data) as an object (not unlike a data dictionary) and then generate an object to hold the data itself. The business model and its tool to pull data from the RDMS, used in the ROLAP case study were more difficult to master technically but it was problems with the data itself, not technical issues, that caused the most difficulty. In both cases, the visual OO development environment allowed rapid development of a GUI user interface to the multi-dimensional data giving the end-user full access to the usual OLAP capability.

Specific Problems in each Case Study

The MOLAP case study encountered two problems. Firstly the literature seemed divided on whether requirements for an EIS should be based on individual executive's critical success factors (CSFs) and key performance indicators (KPIs) or on those of the organisation as a whole. The experience of this case supports the adoption of the individual approach as it was communication with a single manager which established

the required facts and dimensions for the dimensional data model. Observations confirmed that the requirements were specific to this manager and could be different for other managers (Veeraraghavan, 1998) suggesting that an EIS should be tailored to the needs of individual managers even if this increases the time spent on development work.

A second fundamental problem was how to deal with dimensions whose values change over time, for example districts are added, amalgamated or split. Unanticipated changes to the attributes of the dimensions required inherent changes to the dimensional data model and a reload of much of the data. For a small system like this prototype this was not a huge problem but as some system holding gigabytes of data this could be a logistical nightmare and no obvious solution presented itself.

The ROLAP case study was made difficult by the complexities of the data transfer process and the inherent limitations of the tool to do this. It transpired that odd features of the data such as numerical dimension values of identical values within different dimensions, made the automation of this process far from simple. Contact with other users of the commercial product confirmed that this was a common problem and that most did not use the tool preferring instead to export data from the RDDMS to a comma delimited text file and then write a program to read it into the MDDB. However the student/developer observed that the most significant problem faced, when setting up a MDDB from source data located in large organisational database, was one of data cleansing (Dodds, 1998). The bottom-up approach revealed flaws in the data from the overall business perspective that may not concern the transaction processing perspective: for example, different collection times for various data values that make up a KPI.

In each case study there was one person in the organisation with whom the student-developer had most fruitful communication. The first case study using the top-down approach had most contact with the business manager who acted as executive sponsor of the project. The most pressing need was to understand the business imperatives for the system. In the second case using the bottom-up approach most contact was with the database manager familiar with the relational structure from which the data was being extracted.

Conclusion

It is significant that in the top-down approach used in the first (MOLAP) case study there was more focus of the meaning of the information delivered by the MDDB than on the source data quality. In the second (ROLAP) study using a bottom-up approach there was more concern for source data quality not the business meaning of the information in the MDDB. It is suggested that a MOLAP application is most likely to use the top-down approach while a ROLAP development would use the bottom-up approach. It would seem that there are advantages and disadvantages in both approaches.

The results of the first case study show that the multi-dimensional view of data appeals to business managers and adds significantly to the manager's understanding of the state of the enterprise. One way to ensure that this is successful, is to choose facts and dimensions for the MDDB from an analysis of the business needs of the manager who will use the information provided by the system, without concern for how, and from where, data is provided to the system. Although this is important it should not dictate requirements for an EIS.

From the second case study is can been seen that extracting meaningful multi-

dimensional data from a large organisational database is difficult particularly when, as is usually the case, the source data was not designed for that purpose. The differences between relational and dimensional data modelling are significant so that the migration of data from a RDB to an MDDB is a challenging problem. However the migration is made much more difficult by problems of data cleansing where detail required for the dimensional model is missing or ambiguous. The case study highlighted the fact that data cleansing can be mammoth and complex task and there is little guidance in the literature on how to go about it. .

Suggestions for Future Research

This work confirms the value of previously confirmed practices for successful EIS such as the use of prototyping and the existence of executive and operating sponsors (Watson et al., 1993). However the following are research questions, arising from this paper, that are worthy of further study.

The skills and expertise required for dimensional modelling

Research Question 1. Is familiarity with the data and business problem more significant than traditional database expertise?

Tools and methods for data cleansing

Research Question 2. Can business information needs for an EIS be anticipated when building OLTP systems?

Mixing MOLAP and ROLAP approaches

Research Question 3. Can the top-down approach of MOLAP, with its focus on the business needs, and the bottom-up approach of ROLAP, with is focus on extracting the underlying data, be combined into a comprehensive methodology for EIS development?

References

Armstrong D.A. (1990) How Rockwell Launched its EIS, Datamation March 1.

Baum, D., (1996), Data Warehouse, Building Blocks for the Next Millennium, Oracle Magazine, Mar/Apr, pp. 34-43.

Barrow (1992) Implementing an Executive Information System, in Watson, Rainer, Houdeshel (eds) Executive Information Systems, Wiley.

Brancheau J.C. Wetherbe, J.C., (1986) Information Architectures: Method and Practice Information Processing and Management, 22/6, pp 453-464.

Burkan (1991) Executive Information Systems: from Proposal Through Implementation, Van Nostrand Reinhold.

Codd E., Codd S., Salley C. (1993) Providing OLAP to User-Analysts: an IT Mandate. Comshare.

Creeth R. and Pendse N. (1995) The OLAP Report, Business Intelligence

Dodds D. (1998) Towards an Understanding of Current Multi-dimensional OLAP Issues, Internal Report, Department of Business Systems, University of Wollongong, Australia.

Frank M. (1994) A Drill Down Analysis of Multi-dimensional Databases, DBMS, July.

Gentia (1998) OLAP for the Enterprise. http://www.gentia.com/products/gs_eolap.htm

Hasan, H and Gould E (1994) EIS in the Australian Public Sector. Journal of Decision Systems Vol. 3 No. 4, pp 301 - 319

Hasan, H and Hasan S (1997) Computer Based Performance Information For Executives Australian Journal of Public Administration, 56(3) pp. 24-29.
Hasan, H. and Lampitsi, S. (1995) Executive Access to Information in Australian Public Organisations, Journal of Strategic Information Systems 4/2.213-223.
Kimball R. (1994) DBMS Interview: The Doctor of DSS, DBMS Magazine July 1994.
Kimball R. (1997) A Dimensional Modelling Manifesto DBMS Online, www.dbmsmag.com/9708d15.html
Lazer (1996) The Data Breakthrough, LAN Magazine July.
Pokorny J. (1998) Conceptual Modelling in OLAP, Proceeding of ECIS'98, Aix-en-Provence, 273-288.
Rockart J. and De Long D. (1998) Executive Support Systems, Dow Jones-Irwin.
Shanks G.G. O'Donnell P.A. Arnott D.R. (1997) Data Warehousing: A preliminary Field Study Proceedings of the 8th Australasian Conference on Information Systems, Adelaide, 350-365
Shanks, G.G. and O'Donnell, P. (1998) Designing a Data Warehouse: Combining entity Relationship and Dimensional Modelling in D. Arnott, P. O'Donnell and G. Shanks (eds.) Effective Management Support Systems, DSS Laboratory, School of Information Management and Systems, Monash University.
Thierauf (1991)
Veeraraghavan S.R. (1998) Development and Application of a Methodology for Executive Information Systems, Internal Report, Department of Business Systems, University of Wollongong, Australia.
Volonino and Watson (1992) The Strategic Business Objectives Method for Guiding Executive Information Systems Development in Watson, Rainer, Houdeshel (eds) Executive Information Systems, Wiley.
Watson H J, Rainer R K and Koh C E (1993) "Executive Information Systems: A Framework For Development And A Survey of Current Practices" in Sprague R and Watson H J (eds), Decision Support Systems: Putting Theory Into Practice, 3rd Edition, Prentice Hall, Englewood Cliffs, NJ.