

Integrated Decision Support: A Data Warehousing Perspective

Salvatore T. March
Owen Graduate School of Management
Vanderbilt University
Sal.March@owen.vanderbilt.edu

Alan R. Hevner
College of Business Administration
University of South Florida
ahenvner@coba.usf.edu

Extended Abstract

1. Introduction

Successfully supporting managerial decision making has become critically dependent upon the availability of integrated, high quality information organized and presented to managers in a timely and easily understood manner. Data warehouses have emerged to meet this need. Surrounded by analytical tools and models, data warehouses have the potential to transform operational data into *business intelligence*; enabling effective problem and opportunity identification and critical decision making, as well as strategy formulation, implementation, and evaluation (Simon 1977). In this paper, we explore the nature of data warehousing for integrated decision support focusing on research issues and their impact on practice.

Conceptually a data warehouse is extremely simple. As popularized by Inmon (2002) and Inmon and Hackathorn (1994) it is a "subject-oriented, integrated, time-invariant, non-updatable collection of data used to support management decision-making processes and business intelligence." A data warehouse is a repository into which are placed all data relevant to the management of an organization and from which emerge the information and knowledge needed to effectively manage the organization. While this is clearly a simplistic and idealistic view of data warehousing, it allows us to begin the investigation of key challenges and research directions for this discipline.

Figure 1 illustrates our view of a data warehouse as a layered architecture. In a layered architecture, each layer is dependent on the services in lower layers. However, each higher layer is independent of the design decisions made in the lower layers. The following sections discuss the responsibilities and research challenges found in the four data warehouse architecture layers. The kernel Content Management layer addresses data capture, instance-level data integration, and data quality, particularly consistency and timeliness. The Integration and Design layer addresses warehouse design and metadata management. Hence it includes the semantic and conceptual integration, logical and physical design, and performance issues. The Use layer addresses information dissemination including the application of analytical and data mining tools, privacy and security, and user training and support. The Evolution layer addresses change management concerns as the data warehouse responds to changing business needs.

2. Content Management

Managing the content of a data warehouse is a daunting task. Modern organizations use a wide variety of distributed information systems to conduct their day-to-day business. These operational systems draw data from a variety of databases that operate on different hardware platforms, use different operating systems and DBMSs, and have different database structures with varying structural, conceptual, and instance level semantics.

Research has successfully addressed many of the hardware, operating system, DBMS, and structural heterogeneities associated with such systems. However, major challenges remain for data warehouse content management. These include identifying and accessing the appropriate data sources, coordinating data capture from them in an appropriate timeframe, assuring adequate data quality, and instance level integration.

A data warehouse serves as a repository for data extracted from diverse operational information systems. The extraction, transformation, and loading (ETL) functions in a data warehouse are considered the most time-consuming and expensive portion of the development lifecycle (Srivastava and Chen 1999). These processes are concerned with the extraction of data from legacy systems, transformation and preprocessing requirements to produce useful, integrated data, and the transportation of the data into the actual data warehouse structures. Often such operational systems were not designed to be integrated and data extracts are performed manually or on a schedule determined by the operational systems. As a result data in the data warehouse may reflect different states of different systems. Data extracted from an inventory system, for example, may not be synchronized with data extracted from order processing or purchasing systems or specialized systems handling "internal transfers." Reports produced from the data warehouse may be inconsistent and unusable, particularly for real-time decision making situations. Coordination mechanisms must be established. However, the pace of business, particularly Web-based applications, may demand operational systems be available '24 x 7' making extracts and synchronization major problems.

Data quality is a major concern for many operational systems as well as data warehouses (Wand and Wang 1996, Jarke et al. 2000). Validation of accuracy, timeliness, completeness, and consistency remain major problems for many organizations even in internal information systems where users are trained and managed by the organization. These problems are multiplied in information systems that are exposed to customers, vendors, and other partners. The result can be a disaster for a data warehouse that depends on such systems for its content. Mechanisms for protecting a data warehouse from poor quality data are crucial. At the same time rejecting data from an operational system due to quality concerns can exacerbate the data synchronization

problems discussed above, particularly when the organization is using the data warehouse to integrate diverse information systems. Methods for monitoring and cleansing data during ETL have been shown to be successful (Berndt et al. 2003) however, more attention to data quality issues in data warehouses is needed.

Instance level data integration has been studied extensively in the context of heterogeneous databases; however, its solution remains elusive. Organizations routinely have multiple entries in operational databases for the *same* entity in the world. Reasons for this are varied, including data entry errors or limitations of existing software (e.g., disallowing multiple locations for a single customer may cause the organization to maintain a customer record for each location). While this problem is prevalent in internal information systems, the emergence of inter-organizational and Web-based systems and the trend toward mergers and acquisitions magnify its importance.

3. Integration and Design

Given that the data from varied sources have been loaded into the data warehouse, the next set of challenges is the determination, representation, and conceptual integration of the data that are "relevant" to the managerial decision making in an organization. Methodologies for these tasks are in their infancy. Current data and dimensional modeling (e.g., star schemas) approaches for data warehouses (Kimball and Ross 2002) focus almost exclusively on data extracted from current or proposed operations.

Certainly it is valuable to identify *dimensions* along which managers can use to "slice, dice, roll-up, and drill-down" on *facts* acquired by operational systems. However, such an analysis fails to provide an adequate foundation for management decision-making or strategy formulation and evaluation. Methodologies are needed that focus on *business intelligence*

(Simon 1977), that phase of decision making in which managers scan the environment for problems and opportunities. These cannot be limited to the operational level of an organization nor can they be limited to activities that are internal to the business. They must include strategy and goals both within the organization and within its competitive marketplace. They must trace strategy to tactical plans and operational implementation defining key performance indicators at each level. They must be integrative yet flexible, identifying and reconciling heterogeneities among data definitions and concepts used in all levels of management concern.

Furthermore, semantic heterogeneities and instance level integration continue to pose enormous challenges. Briefly stated, a semantic heterogeneity exists when data is defined differently by different users. Differences can be as simple as naming conventions or units of measures which can be easily addressed using conversion tables. Frequently they are much more complex involving different criteria for capturing data and different meanings ascribed to captured data. Such differences must be resolved if data from such systems are to be stored in a common data warehouse. The challenge is to integrate data from diverse information systems in the face of organizational or economic constraints that require those systems to remain autonomous, i.e., retain their differences. Important research challenges that must be addressed include: schema integration, schema evolution, and query processing in such a heterogeneous environment (March, Hevner, and Ram 2000).

Clearly the data warehouse must go beyond its current role as a repository of historical data describing the operations and transactions in which the organization has engaged. It must include data describing partners and partnerships, policies and rules of the business, competitors and markets, goals and standards, opportunities and problems, and alternatives and predicted

futures. Methodologies and representational formalisms for this level of analysis are sorely lacking.

4. Use

Organizations are using data warehousing to support strategic and mission-critical applications. Data deposited into the data warehouse must be transformed into information and knowledge and appropriately disseminated to decision makers within the organization and to critical partners in various supply chains. Crucial problems that must be addressed in this area are: the modes of dissemination of information to the end user; the development, selection, and implementation of appropriate analytical and data mining tools; the privacy and security of data; system performance; and adequate levels of training and support.

The human-computer interface is of paramount importance in the data warehouse environment and the primary determinant of success from the end-user perspective. In order to support analysis and reporting tasks, the data warehouse must have high quality data and make those data accessible through intuitive interface technologies. Data warehouse browsing tools provide star query-like access through a flexible menu-based interface, with pull-down menus representing important dimensions. These types of tools are easy to use and support some ad-hoc exploration, but are usually controlled through an administrative layer that determines the data available to end-users. In developing a flexible interface, there is a tradeoff between the ability to express ad-hoc queries and the ease-of-use that results from pre-defined constructs implemented by data warehouse designers and administrators. Of course, SQL can provide an ad-hoc query facility, but requires some care in the data warehouse environment with very large tables and ill-formed queries conspiring to produce some truly awful performance. In addition,

use of SQL by casual users often produces incorrect queries resulting in erroneous results from the data warehouse.

There are a number of commercially available analytical and data mining tools. Online Analytical Processing (OLAP) tools support multidimensional views of the data warehouse. OLAP "cubes" are frequently extracted from the data warehouse and made available to managers for specific decision making situations. Using tools such as ORACLE Discoverer, Cognos PowerPlay, or Business Objects or even Excel spreadsheets managers can "slice, dice, drill-down, and roll-up" instance-level data along pre-defined dimensions. These can be extremely useful for identifying and exploring the causes of problem situations. For example, drilling down on sales for a specific product that has not met its sales goals can help a manager identify which customers or regions are underperforming with respect to that product. However, they are not very effective for generating solution alternatives once the problem is identified nor are they effective in "discovering" relationships within the data that can be used for strategy formulation or implementation.

Data mining and other "knowledge discovery in database" (KDD) tools, on the other hand, are specifically designed to identify relationships and "rules" within the data warehouse. Unfortunately the identified relationships and rules may or may not be useful to management. Often such tools require users to specify the type of relationship or rule sought. For example, a data mining tool could be used to identify "products that are frequently purchased at the same time" or products whose purchase is "dependent on other previously purchased products." Enabling managers and "power users" to indiscriminately search the data warehouse looking for relationships or rules can raise serious privacy and security concerns, particularly when using Web-based tools.

While analytical and data mining tools have become quite powerful they may be too complex and sophisticated for the average "information consumer." Managers who are comfortable with paper-based reports may find the transition to data warehouse tools to be uncomfortable and counterproductive. Key to effective data warehouse use are identifying the right tools for the different types of data warehouse users and providing adequate training and support once those tools have been selected. For a manager whose primary concern is monitoring sales levels over time by product and sales region a simple Excel spreadsheet automatically connected to an OLAP cube may be sufficient. A manager attempting to identify new marketing strategies and pricing schemes may require more sophisticated tools.

Furthermore, the value of the available tools is dependent upon matching the data characteristics to the managerial need. Early data warehouse applications assumed that currency was not a required characteristic for managerial decision making. Hence data warehouses were often "refreshed" from operational databases on a weekly or monthly basis. Given the accelerated pace of business, "active" or "flash" data warehouses are becoming more prevalent. Such data warehouses are updated virtually in parallel with operational databases. This can lead to integrity and consistency problems because data is in a constant state of flux. Analytical results can vary literally from one moment to another.

5. Data Warehouse Evolution

The key challenge in this layer is that the data warehouse must be "designed for change" from the beginning. As business organizations evolve, their information systems and their data warehouses must evolve with them. New data definitions, new instances, and new tools must be accommodated. Version control becomes crucial. Depending on the data warehouse definition even simple analyses can become problematic in the face of evolving business characteristics.

How, for example, can management interpret historical sales comparisons if sales districts are reorganized or customers are transferred from one salesperson to another? How are historical product sales to be analyzed when, due to a merger or acquisition, product lines have been merged? There is very little theory or guidance available for data warehouse managers to make decisions on how to deal with such changes. Change management in data warehouses is an area ripe for research.

6. Conclusions

We have presented and discussed a layered architecture of data warehousing foundations as summarized in Table 1. The four layers - Content Management, Integration and Design, Use, and Evolution - each present significant value to organizations and challenges for researchers and practitioners. The data warehouse architecture enables the capture and integration of data into the data warehouse and the transformation of that data into useful information and knowledge disseminated appropriately to decision makers within the organization.

Table 1. Research Challenges in Data Warehousing for Integrated Decision Support

Architecture Layer	Research Challenges
Content Management	Data Selection Data Capture Extraction, Transformation, and Loading (ETL) Instance-Level Data Integration Data Quality
Integration and Design	Conceptual Data Integration from Heterogeneous Systems Data Warehouse Schema Design Meta-Model Management Business Intelligence Scanning Supply Chain Integration
Use	Data Dissemination Modes Analytical Models and Tools Data Mining Models and Tools End-User Training and Support Real-Time Updating of Active Data Warehouses
Evolution	Design for Change Change Management and Version Control

References

- Berndt, D., Hevner, A., and Studnicki, J., "The CATCH Data Warehouse: Support for Community Health Care Decision Making," *Decision Support Systems*, Vol. 35, June 2003, pp. 367-384.
- Inmon, W. H., *Building the Data Warehouse*, 3rd ed., New York: Wiley, 2002.
- Inmon, W. H. and Hackathorn, R. D., *Using the Data Warehouse*, New York: Wiley, 1994.
- Jarke, M. et al., Eds., *Fundamentals of Data Warehouses*, Springer-Verlag, Inc., Berlin, 2000.
- Kimball, R. and Ross, M., *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition)*, New York: Wiley, 2002.
- March, S. T., Hevner, A., and Ram, S., " Research Commentary: An Agenda for Information Technology Research in Heterogeneous and Distributed Environments," *Information Systems Research*, Vol. 11, No. 4, December 2000, pp. 327-341.
- Simon, H. A., *The New Science of Management Decision*, Upper Saddle River, NJ: Prentice-Hall, 1977.
- Srivastava, J. and Chen, P., "Warehouse Creation – A Potential Roadblock to Data Warehousing," *IEEE Transactions on Knowledge and Data Engineering*, Vol.11, No. 1, January 1999.
- Wand, Y. and Wang, R. Y., "Anchoring Data Quality Dimensions in Ontological Foundations," *Communications of the ACM*, Vol. 39, No. 11, November 1996, pp. 86-95.

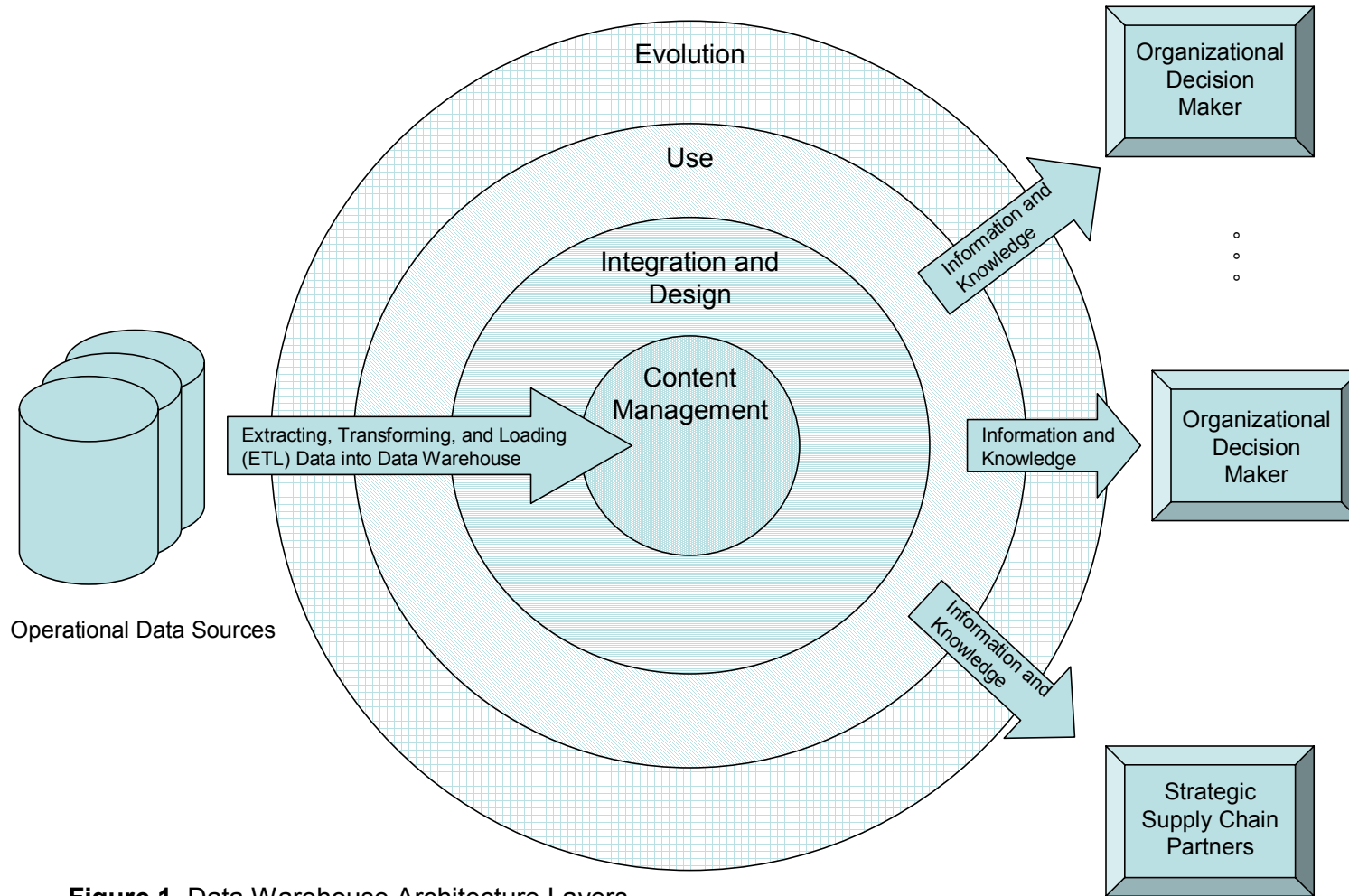


Figure 1. Data Warehouse Architecture Layers