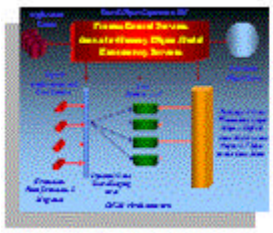

DKMS Briefs

Joseph M. Firestone

DKMS Brief No. Six: Data Warehouses, Data Marts, and Data Warehousing: New Definitions and New Conceptions

Introduction

Bill Inmon's definition of the data warehouse has been dominant since the beginning of the field. It is: "a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process." [1, P. 1] Data Warehousing has lately been undergoing substantial changes in architecture and a broadening of related functional applications. With these changes have come other definitions of the Data Warehouse and evolving conceptions of data warehousing These have been offered with no real attempt to confront other, different definitions or conceptions, or to explore the reasons for disagreements in definitions, and the conceptual commitments or gaps that these definitions imply. This DKMS Brief will explore a number of data warehouse and data mart definitions and their relation to the idea of the Distributed Knowledge Management System (DKMS). [2] It will also analyze the meaning of "data warehousing," in light of changes in data warehousing systems and changes in definitions.

Data Warehouse, Data Mart and the DKMS

Inmon's phrase, taken alone, does not distinguish a data warehouse from a data mart, or enterprise wide data warehouses from process-oriented data warehouses. That is, it does not distinguish subject-oriented, integrated, time-variant, non-volatile data stores of differing scope.

Inmon and his collaborators define a data mart as "a subset of a data warehouse that has been customized to fit the needs of a department." [3, P. 70] They also emphasize that "a data mart is a subset of a data warehouse, containing a small amount of detailed data and a generous portion of summarized data." There is no agreement on this, as there is a strong counter position that atomic data marts are the foundation of the data warehouse. [4, 346-348]

The following types of DSS data stores all fit the characterization "subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process." They represent different concepts, but together they should provide a framework for reasoning about the issue of definition.

- A galactic data warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process about any and all enterprise business processes and departments, and about the enterprise taken as a whole.
- A business process-oriented data warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process about any and all business processes and their interactions with one another and the external world.

- A department-oriented data warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process about any and all departments, and their interactions with one another and with the external world.
- A business process data mart is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process focused on a single business process.
- A departmental data mart is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decision making process focused on a single department.

This framework provides three types of data warehouses and two types of data marts. I think businesses are mostly interested in business process data warehouses and data marts. And that the initial interest in Galactic Data Warehouses has faded, while a focus on departmental data warehouses and data marts is less desirable because it is not consistent with the widely endorsed business process orientation toward increasing productivity. Other recent definitions of data warehouse are focusing on the idea that a simple part-whole relationship exists between these categories and that the union of data marts is the data warehouse.

The union of departmental data marts, however, is not a data warehouse, because this union doesn't necessarily provide management decision support for cross-departmental business processes, or for departmental interactions among themselves and with the external world. Still a departmental data mart is a subset of a galactic data warehouse or a department-oriented data warehouse as defined above.

The union of business process data marts is also not a data warehouse, as Ralph Kimball and his collaborators suggest, [4, Pp. 19, 200-203, 266-271] because this union doesn't necessarily provide management decision support for departments, or for departmental interactions among themselves and with the external world. Still a business process data mart is a subset of a galactic data warehouse or a business process-oriented data warehouse as defined above.

The above definitions of the data warehouse don't preclude the possibility that the data warehouse could be distributed. While insisting that the data warehouse is a unified, integrated logical entity, at the physical level the possibility is there that the data warehouse could be distributed. In other papers, I introduced the DKMS concept [2]. A DKMS is a system that manages the integration of distributed objects into a functioning whole producing, maintaining, and enhancing a business knowledge base. The DKMS must not only manage data, but all of the objects, object models, process models, use case models, object interaction models, and dynamic models, used to process data and to interpret it to produce a business knowledge base. More details on the DKMS are available in [2] and [5].

The DKMS then, is a distributed system of objects. What is the relationship between the Data Warehouse and the DKMS?

A short answer is that a data warehouse is a data store whether distributed or not. While a DKMS is a comprehensive information and knowledge processing system that may include data stores such as data warehouses and data marts, as well clients, application servers, non-DSS data stores, and client front ends. A DKMS can contain a data warehouse and therefore is not strictly comparable to it. What the DKMS is comparable to however, is the data warehousing system -- the system whose components include not only DSS data stores (including the data warehouse and its associated data marts), but also clients and application servers. This comparison will be developed further in discussing the data warehousing process.

Data Warehousing

Moving to "data warehousing" as an activity or process concept, I think the data warehousing field needs to decide whether the term merely refers to establishing, updating, and maintaining a DSS-oriented data store or set of data stores; or whether it refers to a broader process of implementing an information systems application for producing, maintaining, and enhancing an enterprise's knowledge base, including the data store components

of that knowledge base. If we use the term in the broad sense, we must recognize that these days such an application and therefore the data warehousing process it supports, can contain (a) a wide variety of components, and not just front-end clients and back-end database servers, and (b) a variety of functions beyond querying and reporting.

Varieties of Components

Regarding the variety of data warehousing system components, for example, Kimball, Reeves, Ross, and Thornthwaite, distinguish the following main as part of their "Big Shopper Data Warehouse Technical Architecture Model." [4, P. 508]

- Source Systems
- A Data Staging Area with a data staging application server, archive files and other data stores
- A Base Level Presentation Server with a metadata catalog and the data warehouse
- An application server, with a report management server, a ROLAP web interface, a ROLAP service, and a security service
- A Web Server
- A projected OLAP Server Data Mart for Budget Tracking and EIS applications
- A projected Data Mining Server for Customer Scoring, and Purchase Behavior Pattern Analysis Applications
- An end user application client
- An end user advanced analysis client, and
- A web data access client

Many of the same or similar components are identified by Berson and Smith [6, P. 116], who also emphasize the importance of the Information Delivery System in data warehousing architecture, and by Douglas Hackney, in his very interesting "Understanding and Implementing Successful Data Marts." [7, esp. Pp. 61-64]

Indeed, in light of these and other examples, it is hard to avoid the conclusion that the concept of a data warehousing system should recognize the possibility of a range of application server components as part of DW architecture. Figure One, illustrates my own view of the diversity of components in the Data Warehousing System.

Figure One -- The Variety of Data Data Warehousing System Components

Most of the acronyms and abbreviations included in the figure should be familiar. But DDS, ERP, CTS, BPE, and AKM need brief clarifying. DDS refers to the Dynamic Data Store, the data store resulting from the operation of the ETML Server. ERP refers to Enterprise Resource Planning applications such as SAP and BAAN. The issue of the relationship between ERPs and data warehousing is an important current issue in the field, especially since SAP's release of its own data warehousing offering.

CTS is the acronym for Component Transaction Servers. Component Transaction Servers (CTS) such as Sybase's Jaguar CTS, and Microsoft's Transaction Server provide software components with data access and interaction capability. Like Web Application Servers these are stateless, though they do provide broad connectivity and multi-threading, and can play a role in integrating complex data warehousing systems.

BPEs are Business Process Engines, an idea requiring some context to explain. The development of multi-tier distributed processing systems was characterized by the appearance of application servers. Application servers provide services to other components in a distributed processing system by executing business logic and data logic on data accessed from database servers. The class of application servers is sub-divided by Rymer's [8, P. 1ff.] distinction between "stateless" and in-memory server environments. Application Servers with Active in-memory Object Models he calls Business Process Engines, a term similar to Vaskevitch's [9] Business Process Automation Engines.

Application Servers are the primary place where business rules live in the Data Warehousing System. And BPEs mainly perform OLTP and Batch processing under constraints and conditions imposed by these rules.

BPEs may be classified by the processes they support. Some examples of BPEs are: Collaborative Planning Servers, ETML Servers, KDD/Data Mining Servers, Knowledge Publication and Delivery (KPD) Servers, Query and Report creation, modification, scheduling, and Delivery Servers, and Relational On-Line Analytical

Processing (OLAP) Servers. Some of these components have been or will be discussed elsewhere. But it is important to recognize that they are all BPEs because they process business rules, both declarative and procedural, and maintain state in memory.

Finally, AKM is the acronym for Active Knowledge Manager. [5] An AKM is a special kind of BPE providing process control and distribution services, an object model of the Data Warehousing System, and connectivity to all enterprise information, data stores, and applications.

Process Control and Distribution Services include:

- in - memory proactive object state management and synchronization across distributed objects (including business rule management and processing, and metadata management);
- component management;
- workflow management;
- transactional multithreading;
- CORBA and/or COM messaging,

The in-memory Active Object Model and Persistent Object Store Model components of the AKM include:

- Event-driven behavior;
- DKMS-wide model with shared representation;
- Declarative business rules;
- Caching through partial instantiation; and
- A Persistent Object Store for the AKM.

Connectivity Services of the AKM include:

- Language APIs: C, C++, Java, CORBA, COM
- Databases: Relational, ODBC, OODBMS, hierarchical, network, flat file, etc.
- Wrapper connectivity for application software: custom, CORBA, or COM-based
- Applications including all categories mentioned in Figure One.

Changes in data, metadata, business rules or other object methods are arbitrated and synchronized by the AKM. It must perform object and component state management across applications, processing types, and physical platforms through messaging and intelligent agent activities, while providing connectivity to the various data sources and applications within the Data Warehousing System. This is Dynamic Integration (DI). [5] It requires distributed, proactive monitoring and management of changes in objects and components introduced by the various server-based applications in the system.

To perform dynamic integration, the system must:

- look for changes in shared objects and additions to the total pool of objects and relationships,
- alert all system components sharing the objects of such changes, and also
- make decisions about which changes should be implemented in each affected component throughout the system.

It is important that changes in shared objects are propagated and new objects are created in real-time, so that a single view of the Data Warehousing System's object model is maintained. This is why in-memory, proactive operation is so important.

Varieties of Functions

In classifying BPEs by the processes they support I also indicated the variety of functions corresponding to the variety in components: Collaborative Planning, ETML, Knowledge Discovery in Databases (KDD)/Data Mining, Knowledge Publication and Delivery (KPD), Query and Report Creation, Modification, Scheduling, and Delivery, Web Information Access, and Relational On-Line Analytical Processing (ROLAP). Planning,

KDD, KPD, ROLAP and Web Access functions are developments post-dating initial concepts of the Data Warehousing system. The query and reporting management functions have moved from the client to the server. The cumulative effects of these changes is to greatly expand the process content of the data warehousing system, and therefore to change the nature of data warehousing. While previously data warehousing referred to preparing data for an integrated data store, loading data into that store, and then querying the store an off-the-shelf reporting tool; now the term data warehousing can refer to performing the whole range of the above functions.

Conclusion

The continued evolution of data warehousing systems both in the variety of their components and functions, the developing O-O and distributed processing orientation, and the change in definitions to try to accommodate distributed data marts, clarifies the similarity between the Data Warehousing System and the DKMS. The data warehousing system is increasingly like the DKMS in its distributed character, its multi-functionality, and the variety of its components. Two things still distinguish it from the DKMS.

First, though data warehousing systems are evolving toward an object-based integration, they are not there yet. The AKM present in my personal view of data warehousing is still idiosyncratic. In the view of most data warehousing practitioners, the AKM described in this paper, is replaced by a metadata layer having only a fraction of the functionality of the AKM. And second, data warehousing systems are still conceptualized as primarily DSS-based systems, whereas DKMSs recognize no such limitation.

Nevertheless, the trends in the field are clear. With the increasing pressure from ERP and Web-based applications, data warehousing systems will need to integrate transactional applications. And with the further evolution of ETML technology, object models will soon be used to integrate metadata from various stages of the data warehousing process. When these changes come to pass, during the next two years, the evolution of The Data Warehousing System to the DKMS will be complete.

References

[1] W. H. Inmon, "What is a Data Warehouse?" Prism Tech Topic, Vol. 1, No. 1, 1995

[2] I introduced the DKMS concept in two previous White Papers "Object-Oriented Data Warehouse," and "Distributed Knowledge Management Systems: The Next Wave in DSS." Both are available at http://www.dkms.com/White_Papers.htm.

[3] See W. H. Inmon, Claudia Imhoff, and Ryan Sousa, Corporate Information Factory (New York, NY: John Wiley & Sons, 1998).

[4] Ralph Kimball, Laura Reeves, Margy Ross, and Warren Thornthwaite, The Data Warehouse Life Cycle Toolkit (New York: John Wiley & Sons, 1998).

[5] Joseph M. Firestone,"Architectural Evolution in Data Warehousing." available at http://www.dkms.com/White_Papers.htm. See also White Papers 7, 9-11, and the DKMS briefs.

[6] Alex Berson, and Stephen J. Smith, Data Warehousing, Data Mining, and OLAP (New York: McGraw-Hill, 1997).

[7] Douglas Hackney, Understanding and Implementing Successful Data Marts (Reading, MA: Addison-Wesley, 1997).

[8] John Rymer, "Business Process Engines, A New Category of Server Software, Will Burst the Barriers in Distributed Application Performance Engines," Emeryville, CA, Upstream Consulting White Paper, April 7, 1998 at http://www.persistence.com/products/wp_rymer.html. See also my "DKMS Brief No. Four: Business Process Engines in Distributed Knowledge Management Systems," available at http://www.dkms.com/White_Papers.htm

[9] David Vaskevitch, Client/Server Strategies (San Mateo, CA: IDG Books, 1993), P. 288

Biography

Joseph M. Firestone is an independent Information Technology consultant working in the areas of Decision Support (especially Data Marts and Data Mining), Business Process Reengineering and Database Marketing. He formulated and is developing the idea of Market Systems Reengineering (MSR). In addition, he is developing an integrated data mining approach incorporating a fair comparison methodology for evaluating data mining results. Finally, he is formulating the concept of Distributed Knowledge Management Systems (DKMS) as an organizing framework for the next business "killer app." You can e-mail Joe at eisai@home.com.

[Up] [Data Warehouses and Data Marts: New Definitions and New Conceptions]
[Is Data Staging Relational: A Comment]
[DKMA and The Data Warehouse Bus Architecture]
[The Corporate Information Factory or the Corporate Knowledge Factory]
[Architectural Evolution in Data Warehousing]
[Dimensional Modeling and E-R Modeling in the Data Warehouse]
[Dimensional Object Modeling] [Evaluating OLAP Alternatives]
[Data Mining and KDD: A Shifting Mosaic]
[Data Warehouses and Data Marts: A Dynamic View]
[A Systems Approach to Dimensional Modeling in Data Marts]
[Object-Oriented Data Warehousing]