

Selecting a Data Warehouse Architecture

David Septoff, SAS, Rockville, MD
Kevin Simmons, SAS, Rockville, MD

Abstract

Data Warehouse projects have provided end-users with extraordinary returns on investment. Maintaining those extraordinary returns requires an architecture that is robust, flexible and scalable. Careful thought must be given to the selection of the data warehouse and decision-support architecture. A mistake in this area often leads to the deployment of an application that cannot provide long-term support for the data and analytical needs of the end-user community.

Selecting the most appropriate architecture requires examination of a number of factors. Some of those factors are rooted in technical constraints, while others are rooted in how end-users are organized. There are also political issues that must be noted during selection of the architecture. The first step in selecting an architecture is to understand your decision-support needs. The second step is to understand the available architectures. This paper covers some of the most common architectures including:

- Enterprise Data Warehouse (EDW)
- Data Mart
- Operational Data Store (ODS)
- Hybrid Data Warehouse (HDW).

This paper also offers some helpful hints for selecting the most appropriate architecture for your decision-support needs.

This paper will be of interest to anyone who is currently in the planning stages of a data warehouse project. It is assumed that the reader understands the value of data warehousing as it relates to decision-support activities.

Introduction

Data Warehouse development teams are not always given the time to carefully plan and evaluate design decisions. Although this is not optimal, the data warehouse architecture can be determined, or modified

after implementation begins. Waiting until after implementation begins usually increases the amount of rework. Most developers will admit that it is more time consuming and difficult to do rework than to do the job right the first time. For this reason, there is little doubt about the value of a carefully architected data warehouse and decision-support environment. There has been, however, a great deal of debate surrounding the "best" architecture to deploy. The selection of the "best" architecture will be driven by such factors as: available technology, business requirements, commitment to the development effort, skill level of the developers, and the availability of resources.

There are several architectures, which can be supported via the SAS @ System as well as products from other vendors. These architecture options include: Enterprise Data Warehouse (EDW), Data Mart, Operational Data Store (ODS) and Hybrid Data Warehouse (HDW). There are advantages as well as disadvantages for each architecture.

The goal of this paper is to briefly describe each architecture, highlight its key components, and provide the reader with items to consider during the selection of the "best" architecture for their data warehouse project.

What is a Data Warehouse?

The main purpose of the data warehouse is to provide end-users with access to data that can be used in decision-support activities. Bill Inmon (the father of data warehousing) defines the data warehouse as follows: "A data warehouse is a subject oriented, integrated, non-volatile, and time variant collection of data in support of management's decisions."¹ If you extend Inmon's definition to include a collection of data

¹ Inmon, W.H., *Building the Data Warehouse*, New York: Wiley Computer Publishing, 1996, Second Edition

that has physically separated from the source system and passed through quality assurance processes, you have the SAS Institute's definition of a data warehouse.

To better understand Inmon's definition of the data warehouse, let's compare the data warehouse and an online transaction processing (OLTP) system. *See Table 1.* OLTPs are designed to collect and process transactional information that supports routine, day-to-day business operations. OLTP processing includes the addition, update and deletion of extraordinary volumes of data. A common requirement for OLTPs is that data be processed with sub-second response time. Sub-second response time requires that developers focus on achieving optimal performance and system throughput. Support for this type of computing environment usually involves data structures that are entity-relationship based rather than subject oriented. This type of computing environment also puts pressure on developers to limit data redundancy and maximize access speed for small subsets of data.

<i>Data Warehouse</i>	<i>Online Transaction Processing System</i>
Read only access	Read add, update, delete operations
Ad-hoc data access	Predefined data access
Denormalized data model	Highly normalized data model
Historical data	Current data
Summarized and detail level data	Detail level data
Flexible response time	Sub-second response time
Subject oriented	Process oriented

Table 1: Data Warehouse and OLTP Comparison

Data Warehouse Architectures

What is a Data Warehouse Architecture?

The architecture for a data warehouse consists of more than software and the physical storage of data. It should be viewed as an integration of technology and processes. A data warehouse architecture consists of several layers:

- **Conceptual layer** – The conceptual layer includes industry and organizational aspects, corporate vision, a clear description of the problem to be solved and a deployment strategy. The primary output from this layer is a list of data warehouse subjects and a data model.

- **ETL layer** – Software is required to extract data from the source systems, format the data for consistency and accuracy, integrate the data and load the data into the data warehouse. This process is referred to as extraction, transformation, and loading (ETL). Metadata (data about data) is a critical piece of this layer.
- **Management layer** – The management layer facilitates day-to-day support activities for the data warehouse. This support includes archiving data, backing up databases, maintaining system security and job scheduling. Metadata is also a critical piece of this layer.
- **Exploitation layer** – End-users need a facility to run queries against the data warehouse or data mart. The query tool is usually client/server-based or web-based. Query tools need access to metadata as well as security information.

The Enterprise Data Warehouse

The Enterprise Data Warehouse (EDW) will usually support all (or most) of the corporation where there are requirements for flexible data access and usage across departments or business units. *See Figure 1.* The EDW will be a common repository that is leveraged by multiple business units for decision-support activities and its design and construction will be based on the needs of the entire corporation. All the data in the EDW will be based on a common set of business rules, business dimensions, key performance indicators and metrics.

There is no requirement for the EDW to be centralized. The term "enterprise" serves to communicate the depth and breadth of the data warehouse. The EDW can be centralized to a single location or distributed across the enterprise. In either case, the EDW must be managed by the Information Systems (IS) department. *Management* issues for the EDW should not be confused with *control* issues. Distributed locations should be allowed to control what data goes into the EDW, when it is updated, and who has access to the data. Managing the implementation of these decisions should fall to a group with responsibility for the corporate infrastructure, typically IS.

Data for the EDW usually originates from several operational systems. Via batch processes, the source data is extracted from the operational systems, scrubbed, integrated and transformed to meet the requirements of end-users. The data is then loaded into the repository and made available to the end-user community.

The major benefit of the EDW architecture is that it facilitates access to an enterprise-wide view of the data. The major disadvantage of the EDW architecture is that it can be very time consuming, resource intensive and costly to deploy. Before selecting the type of architecture, the data warehouse development team should be certain that access to enterprise-wide data is a requirement.

Enterprise Data Warehouse

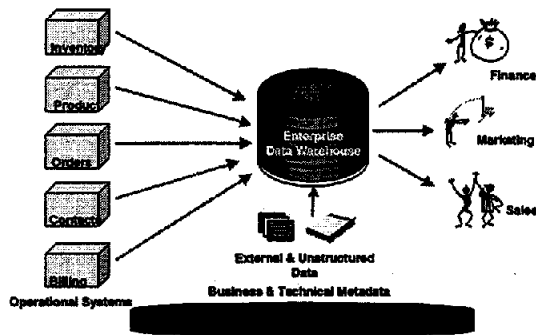


Figure 1: Enterprise Data Warehouse

The Data Mart

A data mart can be described as a departmentally focused data warehouse. It is built to support the data analysis needs of a single group and is usually fed by an Enterprise Data Warehouse and/or operational systems. Two types of data marts are prevalent today: independent (See Figure 2) and dependent.

Independent data marts are stand-alone repositories that are controlled by a single department or business unit. Quite often, a data mart has no connection with other data marts that reside in other business units. Even metadata (data about data) for the data mart may be independent from metadata captured for data marts used by other business units.

Source data for the independent data mart is extracted from operational systems as well as systems that may be external to the business unit for which the data mart is constructed. Source data may also be extracted from an Enterprise Data Warehouse if one exists.

The major benefit for the independent data mart is that a single business unit or department could manage the resources and personnel required for its deployment. This typically has a minimal impact on IS resources and can result in a very fast deployment. The major disadvantages for the independent data mart are that it will only be accessible by the business unit for which it was deployed and the lack of integration constrains the breadth of data to be analyzed. Before selecting this type of architecture, the data warehouse development team should be certain that access by a single business unit is a requirement.

Dependent data marts are interconnected repositories that have been deployed in a distributed fashion. The dependent data marts may be deployed by individual business units or departments, but can be integrated to provide end-users with an enterprise-wide view of data. If the dependent data marts are designed properly, they may be the first steps toward an Enterprise Data Warehouse. A proper design mandates that the dependent data marts share a common set of business rules, business dimensions, key performance indicators and metrics.

Source data for the dependent data mart is extracted from operational systems as well as systems that may be external to the business unit for which the data mart was constructed. Source data may also be extracted from an Enterprise Data Warehouse if one exists.

The major benefit for the dependent data mart is that it supports an enterprise-wide view of data without requiring the IS resources need to build an Enterprise Data Warehouse. The major disadvantage for the dependent data mart is that it has more integration requirements and greater complexity as compared to the independent data mart.

Independent Data Marts

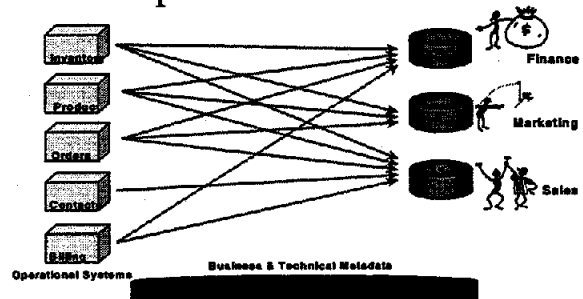


Figure 2: Independent Data Marts

The Operational Data Store

The Operational Data Store (ODS) combines the characteristics of a transaction-processing environment with the characteristics of an integrated decision-support environment. See Figure 3. It provides end-users a consolidated view of relatively current data. Sir stated, the ODS is a copy of operational data that is for operational reporting. Operational reporting reflects the most current available data. It is often used to monitor the day-to-day (or minute-to-minute) activity of a corporation at the transaction level. Very short-term, immediate, and tactical decisions are made using the ODS.

There are four classes of ODSs. Each is distinguished from the others by the frequency with which additional data is loaded and design complexity. The four classes of ODSs are as follows:

- Class I - Updates from operational data sources are done in a real-time or near real-time fashion. Very little data integration or transformation is performed. The level of design complexity for the Class I ODS is low to moderate.
- Class II - Updates from operational data sources are stored and forwarded to the ODS every 1-4 hours. Data is integrated and transformations have been applied. The level of complexity for the Class II ODS is greater than that of the Class I ODS because integration and transformation routines must be developed and great care must be taken to complete data loads within a very short timeframe.
- Class III - Updates from operational data sources are done every 24 hours. Significant data integration and transformation is performed. This type of ODS is relatively easy to build. The level of complexity for the Class III ODS may be less than that of the Class II ODS because there is a larger window for loading data.
- Class IV - The Class IV ODS is fed from a data warehouse and provides end-users with instantaneous access to strategic information. This ODS accepts aggregated analytical data from the data warehouse. This may be the most useful, but most complex ODS to construct. The Class IV ODS is easier to build if the data warehouse is already in place.

The major benefit for the ODS is that it supports an enterprise-wide view of current data. The major

disadvantage for the ODS is the overall design complexity and cost of deployment.

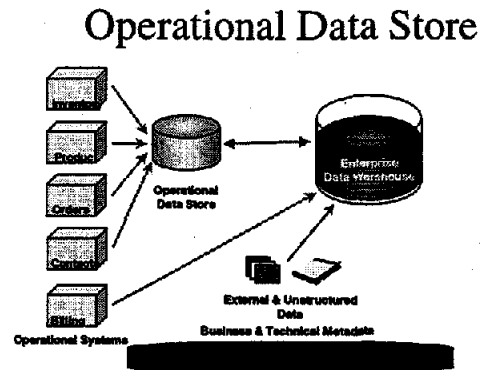


Figure 3: Operational Data Store

The Hybrid Data Warehouse

For years, there has been a debate over the merits of deploying an Enterprise Data Warehouse (EDW) versus deploying data marts. Both approaches have obvious disadvantages as well as advantages and appear to be mutually exclusive. When data marts exist without sharing a common set of business rules, it may be difficult to combine those data marts in an effort to deploy an EDW. Even for corporations with deep financial resources and endless patience, deploying an EDW can be risky and consume a vast amount of resources. One method of obtaining the best of both worlds is the deployment of a Hybrid Data Warehouse (HDW). See Figure 4. The HDW provides end-users with a strategic information repository that is useful for the corporation as well as individual departments or business units.

Creating the HDW requires that the EDW and the departmental data marts be designed in unison. It is not necessary to create a fully attributed, enterprise-wide data model before construction begins. It is necessary, however, to identify and set priorities for the major areas that will eventually be included in the total solution. The initial cut of the enterprise-wide data model must reflect all key fields and how those fields relate to one another. The low-level design and data modeling occurs at the data mart level. The naming conventions, data types and field sizes are clearly defined for each data mart. These conventions become the de facto standards for building the entire decision-support environment and will serve as

the foundation for a fully attributed, enterprise-wide data model.

The HDW can be deployed for organizations or business units that can commit to sharing common business rules, business dimensions, key performance indicators and metrics. If the business units have made substantial investments in OLTP systems, the HDW may be the optimal approach for maximizing that investment without compromising immediate business needs. These represent major advantages for the HDW architecture. The major disadvantage for the HDW is that discipline is required during deployment of each data mart. The design of each data mart must follow the standards established during deployment of previous data marts.

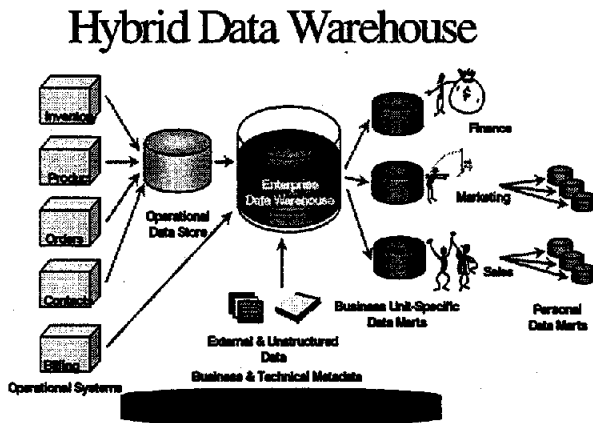


Figure 4: Hybrid Data Warehouse

Selecting an Architecture

Selecting the most appropriate architecture depends on a number of technical and political factors. These factors can be categorized as follows:

- Ownership issues
- End-users issues
- Technical requirements.

Ownership Issues

Requirements for ongoing support for the data warehouse may weigh heavily on the type of architecture that is selected. If IS resources are scarce, an architecture built around data marts should be given

careful consideration. When evaluating architecture alternatives, the data warehouse team should examine the following ownership issues:

Ownership of the data warehouse development group – The data warehouse development group often belongs to an organization that also concentrates its efforts on the development of transaction processing systems. This can be a great benefit to the development effort because the group that provides support for both types of systems is usually familiar with the source data that is being fed into the data warehouse. A downside to this arrangement is that the support group may be driven to respond to the crisis at hand. If it comes to choosing between supporting the transaction processing system and the data warehouse, the transaction processing system usually gets the attention.

Ownership of the databases used to house the data – Building a data warehouse is a very iterative activity. This may necessitate relatively frequent changes to database structures. Developers will want to make these changes quickly and won't want to be put on a list of backlog requests for the database administrator (DBA). On the other hand, DBAs can usually help with performance issues.

Ownership of data quality issues – Monitoring the quality of data in the warehouse is not a one-time event. It should be an ongoing process that is supported by formal procedures. It is not an easy task and becomes more difficult as more data sources are added. End-users tend to push this task to the data warehouse development team. The data warehouse development team should also be prepared to take responsibility for correcting the invalid data. Selecting the process by which corrections are applied to invalid data can be a politically charged decision. The labor and computing resources required to correct the data often drive the selection of a data correction process.

End-User Issues

End-user issues cannot be ignored when selecting the data warehouse architecture. These issues often drive many of the technical requirements that the data warehouse must support. The following issues should be given careful consideration when the architecture alternatives are evaluated:

Types of end users – Not all end-users are created equal. There are categories of users with very distinct needs. Some end users will only use the data warehouse to monitor key performance indicators. Others will scan large volumes of data to confirm a hypothesis. Another

group may look for previously undiscovered patterns in the data.

Users may not wish to abandon existing databases – Many end-user groups have created their own databases and decision-support environment because their IS department has been unwilling or unable to provide adequate support. After the data warehouse is up and running, those end-user communities may be skeptical about whether IS will be able to support their reporting needs.

Support the needs of the many or the needs of the most demanding – For some data warehouses, there may be a small group of infrequent users that require huge amounts of data and/or highly complex reports. Their needs may go far beyond the needs of the every day user of the data warehouse. Satisfying those needs may consume vast amounts of time, money and computing resources. On the other hand, these users may be the most vocal or may contribute the largest amount of money to the project. They may also have a need that is highly beneficial to the business.

Legitimate access to the data – Most data warehouses are built with data that is created, maintained or needed

Configuration management	Data retention
Data and metadata archive	DBMS
Data and metadata backup	Load window
Data quality	Metadata management
Data recovery	Physical data model

Summary

There is an old saying that goes something like: “if it walks like a duck and talks like a duck, etc. it is probably a duck”. It is not quite that simple to identify the best architecture for your data warehousing and decision-support needs. After your data warehouse project has been approved, selecting the most appropriate architecture may be the most difficult decision you have to make. Great care must be taken to evaluate the available alternatives. Selecting the best architecture depends on a number of factors. Some of those factors are rooted in technical constraints, while others are rooted in how end-users are organized. There are also political issues that must be noted during selection of the architecture.

Four data warehouse architectures seem to hold the most promise for providing end-users with an environment that is flexible, scalable and maintainable. Those architectures include:

- Enterprise Data Warehouse (EDW)

by multiple groups. One group may not want other groups to see their data. One location may not want other locations to see their data. Some groups may be allowed to see aggregated data, but restricted from viewing detail level data.

The need for current data – It is not uncommon for end-users to ask for access to all the available data as soon as it is available. The data warehouse development team may assume that data needs to be updated far more often than makes business sense. If the data warehouse architecture is in place to support day-to-day activities, it should be updated daily. If it is in place to support week-to-week activities, it should be updated weekly.

Technical Requirements

Technical requirements usually carry the most weight during the evaluation of a data warehouse architecture. Corporate computing standards and the data warehouse tools may impose constraints on the data warehouse architecture. The following is a list (although not exhaustive) of technical requirements that should be considered when selecting the data warehouse architecture:

Response time
Scalability
Security

- Data Mart (dependent and independent)
- Operational Data Store (ODS)
- Hybrid Data Warehouse (HDW).

Each architecture alternative has its advantages and disadvantages.

The major benefit for the EDW architecture is that it facilitates access to an enterprise-wide view of the data. The major disadvantage of the EDW architecture is that it can be very time consuming, resource intensive and costly to deploy. Before selecting the type of architecture, the data warehouse development team should be certain that access to enterprise-wide data is a requirement.

The major benefit for the independent data mart is that a single business unit or department could manage the resources and personnel required for its deployment. This typically has a minimal impact on IS resources and can result in a very fast deployment. The major disadvantages for the independent data mart are that it will only be accessible by the business unit for which it was deployed and the lack of integration constrains the breadth of data to be

analyzed. Before selecting this type of architecture, the data warehouse development team should be certain that access by a single business unit is a requirement.

The major benefit for the dependent data mart is that it supports an enterprise-wide view of data without requiring the IS resources need to build an Enterprise Data Warehouse. The major disadvantage for the dependent data mart is that it has more integration requirements and greater complexity as compared to the independent data mart.

The major benefit for the ODS is that it supports an enterprise-wide view of current data. The major disadvantage for the ODS is the overall design complexity and cost of deployment.

The HDW can be deployed for organizations or business units that can commit to sharing common business rules, business dimensions, key performance

indicators and metrics. If the business units have made substantial investments in OLTP systems, the HDW may be the optimal approach for maximizing that investment without compromising immediate business needs. These represent major advantages for the HDW architecture. The major disadvantage for the HDW is that discipline is required during deployment of each data mart. The design of each data mart must follow the standards established during deployment of previous data marts.

Each of the four major data warehouse architecture alternatives depends on solid technology, experienced resources and proven deployment practices to be successful. Fortunately, the SAS @ System and the SAs Institute's Rapid Warehousing Methodology can support the deployment of an Enterprise Data Warehouse, Data Mart, Operational Data Store and Hybrid Data Warehouse.

Table 2 provides a quick look at potential evaluation criteria for each data warehouse architecture alternative.

	Operational Data Store (ODS)	Dependent Data Mart	Enterprise Data Warehouse (EDW)	Hybrid Data Warehouse (HDW)
Audience	Analysts and administrators	Business analysts	Analysts and executives	Analysts and executives
Complexity	Medium to high	Simple to medium	Medium to high	Simple to medium
Data access	Limited volume	Relatively small	Large	Large
Data content	Current data	Historical data	Historical data	Historical data
Data granularity	High	Low	High	High
Data model	Normalized	Denormalized	Normalized	Normalized
Data sources	Many	Few	Many	Many
Data usage	Highly structured queries	Structured queries	Unstructured queries	Unstructured queries
DSS support	Tactical activities	Strategic decision making	Strategic decision making	Strategic decision making
Management	Centralized IT	Decentralized departments	Centralized IT	Centralized IT

Criteria	Greenfield Info Solutions	Greenfield Info Solutions	Greenfield Info Solutions	Greenfield Info Solutions
Architecture	Low	Medium	High	High
Nature of data	Dynamic	Static	Static	Static
Requirements	Relatively stable	Unstable	Highly unstable	Highly unstable
Scalability	High	Low	High	High
Technical expertise required	Higher	Lower	Higher	Higher

Table 2: Data Warehouse Architecture Evaluation Criteria

About the Authors

David Septoff is the director of the Integrated Solutions Consulting department at SAS

He can be reached at:
Integrated Solutions Consulting
SAS, Rockville, MD
301-881-8840
David.Septoff@sas.com

Kevin G. Simmons is the manager of the Data Warehouse Practice at SAS

He can be reached at:
Integrated Solutions Consulting
SAS, Rockville, MD
301-881-8840
Kevin.Simmons@sas.com

References

Greenfield, Larry, LGI Systems Inc.,
"The Data Warehousing Information Center,"
<http://www.dwinfocenter.org>

Hackney, Douglas, *Understanding and Implementing Successful Data Marts*
Addison Wesley Longman, 1997

Hill, Harjinder S. & Rao, Prakash C. *The Official Guide to Data Warehousing*
Que Corporation, 1996

Inmon, W.H., *Building the Data Warehouse*,
Wiley Computer Publishing, 1996

Kimball, Ralph, *The Data Warehouse Toolkit*,
Wiley Computer Publishing, 1996

SAS Institute Inc., "A SAS Institute White Paper: The SAS® Data Warehouse", Cary, NC:
SAS Institute Inc., Available through SAS
Institute's Sales and Marketing Division

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other Countries. ® indicates USA registration.

Acknowledgments

The authors wish to acknowledge Tony Brown, Modibo Coulibaly, Allen Cunningham, Jack May, and Fred Wilds for their contributions to this paper.

