Data Warehousing

The data warehouse is the heart of the architected environment, and is the foundation of all DSS processing. The job of the DSS analyst in the data warehouse environment is massively easier than in the classical legacy environment because there is a single integrated source of data (the data warehouse) and because the granular data in the data warehouse is easily accessible.

A data warehouse is a large database built from the operational database that organizes all the data available in an organization, makes it accessible & usable for the all kinds of data analysis and also allows to create a lots of reports by the use of mining tools.

"A data warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data in support of management's decision-making process."

Subject Oriented

The subject orientation of the data warehouse is shown in Figure 1. Classical operations systems are organized around the applications of the company. For an insurance company, the applications may be auto, health, life, and casualty. The major subject areas of the insurance corporation might be customer, policy, premium, and claim. For a manufacturer, the major subject areas might be product, order, vendor, bill of material, and raw goods. For a retailer, the major subject areas may be product, sale, vendor, and so forth. Each type of company has its own unique set of subjects.

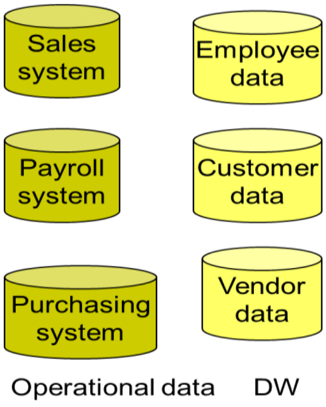

***Figure 1** The Issue of Subject orientation*

Integrated

The second salient characteristic of the data warehouse is that it is integrated. Of all the aspects of a data warehouse, integration is the most important. Data is fed from multiple disparate sources into the data warehouse. As the data fed it is converted, reformatted, resequenced, summarized, and so forth. The result is that data—once it resides in the data warehouse—has a single physical corporate

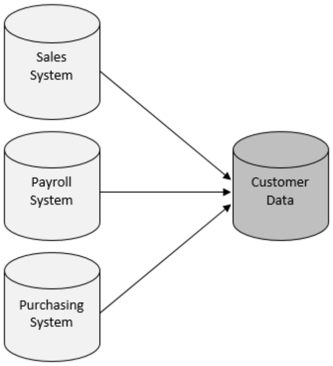

***Figure 2** The issue of integration*

image. Figure 2 illustrates the integration that occurs when data passes from the application-oriented operational environment to the data warehouse.

Nonvolatile

The third important characteristic of a data warehouse is that it is nonvolatile. Figure 3 illustrates nonvolatility of data and shows that operational data is regularly accessed and manipulated one record at a time. Data is updated in the operational environment as a regular matter of course, but data warehouse data exhibits a very different set of characteristics. Data warehouse data is loaded (usually en masse) and accessed, but it is not updated (in the general sense). Instead, when data in the data warehouse is loaded, it is loaded in a snapshot, static format. When subsequent changes occur, a new snapshot record is written. In doing so a history of data is kept in the data warehouse.

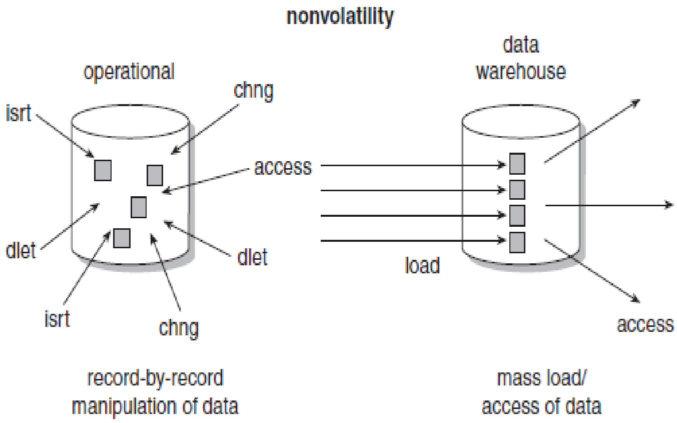

Figure 3 *The issue of nonvalatility*

Time Variant

Time variance implies that every unit of data in the data warehouse is accurate as of some one moment in time. In some cases, a record is time stamped. In other cases, a record has a date of transaction. Different environments have different time horizons. A time horizon is the parameters of time represented in an environment. The collective time horizon for the data found inside a data warehouse is significantly longer than that of operational systems. A 60-to-90-day time horizon is normal for operational systems; a 5-to-10-year time horizon is normal for the data warehouse. As a result of this difference in time horizons, the data warehouse contains much more history than any other environment

Differences Between Operatonal Database and Data Warehouse

	Data Warehouse	Operational Database
Purpose	Analysis, Decision making	Day-to-Day
Supports	OLAP	OLTP
Users	Managerial community	Clerical Community
Data Model	Multi-dimensional	Relational
Age of Data	Current and time series	Current and real time
Data Modification	Read/Access only	Insert, update and delete
Types of Data	Static	Dyanamic
Amount of Data per Transaction	Larger	Smaller

Data Mart

A data mart contains a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to specific selected subjects. For example, a marketing data mart may confine its subjects to customer, item, and sales. The data contained in data marts tend to be summarized. Data marts are usually implemented on low-cost departmental servers that are UNIX/LINUX or Windows based. The implementation cycle of a data mart is more likely to be measured in weeks rather than months or years. However, it may involve complex integration in the long run if its design and planning were not enterprise-wide.

Depending on the source of data, data marts can be categorized as independent or dependent. *Independent* data marts are sourced from data captured from one or more operational systems or external information providers, or from data generated locally within a particular department or geographic area. *Dependent* data marts are sourced directly from enterprise data warehouses.

Operational Data Sources

Operational data sources (ODS) exist to support daily operations. The ODS data is cleaned and validated, but it is not historically deep, it may be just the data for the current day. The ODS may also be used as a source to load the data warehouse.

Extract, Transform, Load (ETL)

A data warehouse usually stores many years of data to support historical analysis. The data in a data warehouse is typically loaded through an extraction, transformation, and loading (ETL) process from multiple data sources. Modern data warehouses are moving toward an extract, load, and transformation (ELT) architecture in which all or most data transformation is performed on the database that hosts the data warehouse.

Extract

- Data extraction (retrieval) from source system
- Incremental extract: identifying modified record and extract only those
- Full extract: extract full copy in same format to identify changes

Transform

- Transform data from source to the target
- Conversion to same dimension, units etc.
- Generating aggregates, sorting, validating etc.

Load

- Load the data into temporary data source first and then perform simple transformation into structure similar to one in data warehouse
- Consume as little resource as possible

Data Warehouse Process Managers and Functions

Load Manager: The system components that perform all the operations necessary to support the extract and load process. It fast loads the extracted data into a temporary data store and performs simple transformations into a structure similar to the one in the data warehouse also called ETL (Extract Transform and Load).

Warehouse Manager (Data Manager): It is the system component that performs analysis of data to ensure consistency. The data from various sources and temporary storage are merged into data warehouse by the warehouse manager. The job of backing-up and archiving data as well as creation of index is performed by this manager.

Query Manager: Performs all the operations necessary to support the query management process by directing queries to the appropriate tables. They present the data to the user in a form they understand. In some cases it also stores query profiles to allow the warehouse manager to determine which indexes and aggregations are appropriate.

Data Warehouse Architecture

--- *Refer notes of unit one* ---

Data Warehouse Design

The entity-relationship data model is commonly used in the design of relational databases, where a database schema consists of a set of entities and the relationships between them. Such a data model is appropriate for on-line transaction processing.

A data warehouse requires a concise, subject-oriented schema that facilitates on-line data analysis. The most popular data model for a data warehouse is a multidimensional model. Such a model can exist in the form of ***a star schema***, ***a snowflake schema,*** or ***a fact constellation schema.***

--- *Refer notes of unit two* ---

GUIDELINES FOR DATA WAREHOUSE IMPLEMENTATION

Implementation Steps

1. **Requirements analysis and capacity planning:** In the first step in data warehousing involves defining enterprise needs, defining architecture, carrying out capacity planning and selecting the hardware and software tools. This step will involve consulting senior management as well as the various stakeholders.

2. Hardware integration: Once the hardware and software have been selected, they need to be put together by integrating the servers, the storage devices and the client software tools.

3. Modeling: Modeling is a major step that involves designing the warehouse schema and views. This may involve using a modeling tool if the data warehouse is complex.

4. Physical modeling: For the data warehouse to perform efficiently, physical modeling is required. This involves designing the physical data warehouse organization, data placement, data partitioning, deciding on access methods and indexing.

5. Sources: The data for the data warehouse is likely to come from a number of data sources. This step involves identifying and connecting the sources using gateways, ODBC drives or other wrappers.

6. ETL: The data from the source systems will need to go through an ETL process. The step of designing and implementing the ETL process may involve identifying a suitable ETL tool vendor and purchasing and implementing the tool. This may include customizing the tool to suit the needs of the enterprise.

7. Populate the data warehouse: Once the ETL tools have been agreed upon, testing the tools will be required. Once everything is working satisfactorily, the ETL tools may be used in populating the warehouse given the schema and view definitions.

8. User applications: For the data warehouse to be useful there must be end-user applications. This step involves designing and implementing applications required by the end users.

9. Roll-out the warehouse and applications: Once the data warehouse has been populated and the end-user applications tested, the warehouse system and the applications may be rolled out for the user community to use.

Implementation Guidelines

1. Build incrementally: Data warehouses must be built incrementally. Generally it is recommended that a data mart may first be built with one particular project in mind and once it is implemented a number of other sections of the enterprise may also wish to implement similar systems. An enterprise data warehouse can then be implemented in an iterative manner allowing all data marts to extract information from the data warehouse. Data warehouse modeling itself is an iterative methodology as users become familiar with the technology and are then able to understand and express their requirements more clearly.

2. Need a leader: A data warehouse project must have a leader who is willing to carry out considerable research into expected costs and benefits of the project. Data warehousing projects require inputs from many units in an enterprise and therefore need to be driven by someone who is capable of interaction with people in the enterprise and can actively persuade colleagues. Without the cooperation of other units, the data model for the warehouse and the data required to populate the warehouse may be more complicated than they need to be. Studies have shown that having a champion can help adoption and success of data warehousing projects.

3. Senior management support: A data warehouse project must be fully supported by the senior management. Given the resource intensive nature of such projects and the time they can take to implement, a warehouse project calls for a sustained commitment from senior management. This can sometimes be difficult since it may be hard to quantify the benefits of data warehouse technology and the managers may consider it a cost without any explicit return on investment. Data warehousing project studies show that top management support is essential for the success of a data warehousing project.

4. Ensure quality: Only data that has been cleaned and is of a quality that is understood by the organization should be loaded in the data warehouse. The data quality in the source systems is not always high and often little effort is made to improve data quality in the source systems. Improved data quality, when recognized by senior managers and stakeholders, is likely to lead to improved Support for a data warehouse project.

5. Corporate strategy: A data warehouse project must fit with corporate strategy and business objectives. The objectives of the project must be clearly defined before the start of the project. Given the importance of senior management support for a data warehousing project, the fitness of the project with the corporate strategy is essential.

6. Business plan: The financial costs (hardware, software, and HR), expected benefits and a project plan (including an ETL plan) for a data warehouse project must be clearly outlined and understood by all stakeholders. Without such understanding, rumors about expenditure and benefits can become the only source of information, undermining the project.

7. Training: A data warehouse project must not overlook data warehouse training requirements. For a data warehouse project to be successful, the users must be trained to use the warehouse and to understand its capabilities. Training of users and professional development of the project team may also be required since data warehousing is a complex task and the skills of the project team are critical to the success of the project.

8. Adaptability: The project should build in adaptability so that changes may be made to the data warehouse if and when required. Like any system, a data warehouse will need to change, as needs of an enterprise change. Furthermore, once the data warehouse is operational, new applications using the data warehouse are almost certain to be proposed. The system should be able to support such new applications.

9. Joint management: The project must be managed by both IT and business professionals in the enterprise. To ensure good communication with the stakeholders and that the project is focused on assisting the enterprise's business, business professionals must be involved in the project along with technical professionals.

References

[1] J. Han and K. Micheline, Data Mining: Concepts and Techniques, San Francisco: Elsevier Inc., 2006.

[2] W. Inmon, Building the Data Warehouse, New York: John Wiley & Sons, Inc., 2002.