© 2014 IJIRT | Volume 1 Issue 6 | ISSN : 2349-6002

DATA WAREHOUSE

Sheena Batra, Rakesh Sondal
Dronacharya College of Engineering, Guragon

***Abstract*—Organization's historical data is main source for many business applications, such as mining on the data, planning, statistical analysis or departmental based reporting .These are two important features of data warehouse that are-**

As a storage medium, data warehouse and mart has been widely used to hold this type of the organization information on it; In addition, they are likely to be main decision support technologies because they have ability to support online analytical processing. As a result of these important features of it, it has quite different performance, peak loads, and capacity requirements than online transactional processing system. As a guideline, I lights on the types of the data warehouse architectures with a representative sketches in order to make organization being certain about them. As a storage medium these data warehouses can store information in bulk that's why these houses are very useful to us.

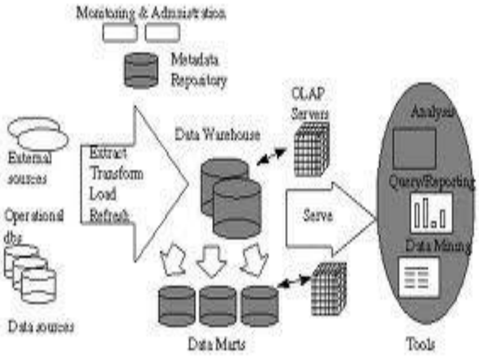
I. INTRODUCTION

Data warehousing is a collection of *decision support* technologies, aimed at enabling the *knowledge worker* (executive, manager, analyst) to make better and faster decisions. The past three years have seen explosive growth, both in the number of products and services offered, and in the adoption of these technologies by industry. According to the *META Group*, the data warehousing market, including hardware, database software, and tools, is projected to grow from $2 billion in 1995 to $8 billion in 1998. Data warehousing technologies have been successfully deployed in many industries: manufacturing (for order shipment and customer support), retail (for user profiling and inventory management), financial services (for claims analysis, risk analysis, credit card analysis, and fraud detection), transportation (for fleet management), telecommunications (for call analysis and fraud detection), utilities (for power usage analysis), and healthcare (for outcomes analysis). This paper presents a roadmap of data warehousing technologies, focusing on the special requirements that data warehouses place on database management systems (DBMSs). A data warehouse is a "subject-oriented, integrated, timevarying, non-volatile collection of data that is used primarily in organizational decision making."1 Typically, the data warehouse is maintained separately from the organization's operational databases. There are many reasons for doing this. The data warehouse supports on-line analytical processing (OLAP), the functional and performance requirements of which are quite different from those of the on-line transaction processing (OLTP) applications traditionally supported by the operational databases.

OLTP applications typically automate clerical data processing tasks such as order entry and banking transactions that are the bread-and-butter day-to-day operations of an organization. These tasks are structured and repetitive, and consist of short, atomic, isolated transactions. The transactions require detailed, up-to-date data, and read or update a few (tens of) records accessed typically on their primary keys. Operational databases tend to be hundreds of megabytes to gigabytes in size. Consistency and recoverability of the database are critical, and maximizing transaction throughput is the key performance metric. Consequently, the database is designed to reflect the operational semantics of known applications, and, in particular, to minimize concurrency conflicts. Data warehouses, in contrast, are targeted for decision support. Historical, summarized and consolidated data is more important than detailed, individual records. Since data warehouses contain consolidated data, perhaps from several operational databases, over potentially long periods of time, they tend to be orders of magnitude larger than operational databases; enterprise data warehouses are projected to be hundreds of gigabytes to terabytes in size. The workloads are query intensive with mostly ad hoc, complex queries that can access millions of records and perform a lot of scans, joins, and aggregates. Query throughput and response times are more important than transaction throughput.

To facilitate complex analyses and visualization, the data in a warehouse is typically modeled *multidimensionally*. For example, in a sales data warehouse, time of sale, sales district, salesperson, and product might be some of the dimensions of interest. Often, these dimensions are hierarchical; time of sale may be organized as a day-month-quarter-year hierarchy, product as a product-category-industry hierarchy.

II. ARCHITECTURE AND PROCESS OF DESIGN



The left most tier (bottom tier) is a **warehouse database server** that is almost always a relational database system. Data from operational databases and external sources are extracted using application program interface known as gateways. A gateway is supported by the underlying DBMS and allows client programs to generate SQL code to be executed by a server. The middle tier is an OLAP server that is typically implemented using a ROLAP that maps operations on multidimensional data to standard relational operations or using a MOLAP that is a special purpose server that directly implements multidimensional data and operations. The right most tier (top tier) is a client, which contains query and reporting tools, analysis tools, and data mining tools.

A data warehouse can be built using a top-down approach, a bottom-up approach, or a combination of both. The top-down approach starts with the overall design and planning. It is useful in cases where the technology is mature and well known, and where the business problems that must be solved are clear and well understood. The bottom-up approach starts with experiments and prototypes. This is useful in the early stage of business modeling and technology development It allows an organization to move forward at considerably less expense and to evaluate the benefits of the technology before making significant commitments. In the combined approach, an organization can exploit the planned and strategic nature of the top-down approach while retaining the rapid implementation and opportunistic application of the bottom-up approach.

The design and construction of the data warehouse consists of the following steps: planning, requirements study, problem analysis, warehouse design, data integration and testing, and finally deployment of the data warehouse.

a. Choose a business process to model, for example, shipments, inventory, sales and the general ledger.

b. Choose the grain of the business process. The grain is the Fundamental, atomic level of data to be represented in the fact table for this process.

c. Choose the dimensions that will apply to each fact table record.

d. Choose the measures that will populate each fact table record.

Once a data warehouse is designed and constructed, the initial deployment of the warehouse includes initial installation, roll out planning, training and orientation.

A. Back-End Tools And Utilities

Data warehouse systems use back-end tools and utilities to populate and refresh their data. **Data Cleaning:**

Data cleaning routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data

Since a data warehouse is used for decision making, it is important that the data in the warehouse must be correct. Some examples where data cleaning becomes necessary are: inconsistent field length, inconsistent descriptions, inconsistent value assignments, missing entries and violation of integrity constraints.

Load

After extracting, cleaning and transforming, data must be loaded into the warehouse. Additional preprocessing may still be required: checking integrity constraints; sorting; summarization; aggregation; and other computations to build the derived tables stored in the warehouse. In addition, load utility also allows the system administrator to monitor status, to cancel, to suspend and resume a load, and to restart after failure with no loss of data integrity.

The load utilities for data warehouses have to deal with much larger data volumes than for operational databases

Refresh

Refreshing a warehouse consists in propagating updates on source data to correspondingly update the base data and derived data stored in the warehouse. There are two sets of issues to consider: when to refresh and how to refresh. Usually, the warehouse is refreshed periodically. The refresh policy is set by the warehouse administrator, depending on user needs and traffic, and may be different for different sources. Refresh techniques also depends on the characteristics of the source and capabilities of the database servers. Replication servers can be used to refresh a warehouse when the sources change.

B. Multidimensional Data Model

Data warehouse and OLAP tools are based on a multidimensional data model. This model views data in the form of a data cube. A data cube allows data to be modeled and viewed in multiple dimensions. Dimensions are perspectives or entities with respect to which an organization wants to keep records. Each dimension has a table associated with it, called a dimension table, which further describes the dimension.

A multidimensional data model is typically organized around a central theme. This theme is represented by a fact table. The fact table contains the names of the facts, or measures, as well as keys to each of the related dimension tables.

C. OLAP Servers

Logically, OLAP servers present business users with multidimensional data from data warehouses or data marts, without concerns regarding how or where the data stored. However, the physical architecture and implementation of OLAP servers must consider data storage issues. Implementation of a warehouse server for OLAP processing includes the following:

Relational OLAP servers - These are the intermediate servers that stand in between a relational back-end server and client front-end tools. ROLAP severs include optimization for each DBMS back end, implementation of aggregation navigation logic, and additional tools and services.

Multidimensional OLAP servers – These servers support multidimensional views of data through array-based multidimensional storage engines. Many MOLAP servers adopt a two-level storage representation to handle sparse and dense data sets: the dense sub cubes are identified and stored as array structures, while the sparse sub cubes employ compression technology for efficient storage utilization.

Hybrid OLAP servers – The hybrid OLAP approach combines ROLAP and MOLAP technology, benefiting from the greater scalability of ROLAP and faster computation of MOLAP. For example, a HOLAP server may allow large volumes of detail data to be stored in a relational database, while aggregations are kept in separate MOLAP store.

D. Metadata Repository

Metadata are data about data. When used in a data warehouse, metadata are the data that define warehouse objects. Metadata are created for the data names and definitions of the given warehouse. Administrative metadata includes all of the information necessary for setting up and using a warehouse; description of the source databases; back-end and front-end tools. Business metadata includes business terms and definitions; ownership of the data. Operational metadata includes information that is created during the operation of the warehouse; monitoring information such as usage statistics, error reports, and audit trails.

Metadata repository is used to store and manage all the metadata associated with the warehouse. The repository enables the sharing of the metadata among tools and processing for designing, setting up, using, operating and administering a warehouse.

Metadata play a very different role than other data warehouse data, and are important for many reasons. For example, metadata are used as a directory to help the decision support mapping of the data when the data are transformed from the operational environment to the data warehouse environment. So, metadata should be stored and managed persistently.

III. RESEARCH ISSUES

We have described the substantial technical challenges in developing and deploying decision support systems. While many commercial products and services exist, there are still several interesting avenues for research. We will only touch on a few of these here.Data cleaning is a problem that is reminiscent of heterogeneous data integration, a problem that has been studied for many years. But here the emphasis is on *data* inconsistencies instead of schema inconsistencies. Data cleaning, as we indicated, is also closely related to data mining, with the objective of suggesting possible inconsistencies.The problem of physical design of data warehouses should rekindle interest in the well-known problems of index selection, data partitioning and the selection of materialized views. However, while revisiting these problems, it is important to recognize the special role played by aggregation.

Decision support systems already provide the field of query optimization with increasing challenges in the traditional questions of selectivity estimation and cost-based algorithms that can exploit transformations without exploding the search space (there are plenty of transformations, but few reliable cost estimation techniques and few smart cost-based algorithms/search strategies to exploit them). Partitioning the functionality of the query engine between the middleware (e.g., ROLAP layer) and the back end server is also an interesting problem. The management of data warehouses also presents new challenges. Detecting runaway queries, and managing and scheduling resources are problems that are important but have not been well solved. Some work has been done on the logical correctness of incrementally updating materialized views, but the performance, scalability, and recoverability properties of these techniques have not been investigated. In particular, failure and checkpointing issues in load and refresh in the presence of many indices and materialized views needs further research. The adaptation and use of workflow technology might help, but this needs further investigation. Some of these areas are being pursued by the research

community33 34, but others have received only cursory attention, particularly in relationship to data warehousing

IV. CONCLUSION

Justifying a data warehouse project can be very difficult. Usually, analysis of the success of the data warehouse project is done considering the financial benefits against the investment. Since most of the educational institutes are nonprofit organizations and service oriented, the evaluation of the usefulness of the data warehouse can be done on the basis of its ability to meet user’s requirements. The academic data which was spread all across different sources has been loaded into single platform. The decision makers can extract information regarding three main components of the institute,namely Employees, Students and Infrastructure. Employee data mart can provide the users with the information such as career growth and attrition rate. Student mart can provide them information related to the student like best outgoing student considering his academic and non academic activities. Information regarding assets such as the investment in a particular financial year can also be accessed. In educational institute, decision makers ask “What are the expected results and benefits?” when making a data warehouse project rather than “What is the anticipated return on investment?”. The data warehouse developed has met their expectations. Benefits of the present project can be more if the Institute has positive approach towards new technologies. They can take micro-level decisions in a timely manner without the need to depend on their IT staff. They can perform extensive analysis of stored data to provide answers to the exhaustive queries to the administration cadre. This helps them to formulate strategies and policies for employees and students. This helps students and Employees in making decisions. They are the ultimate beneficiaries of the new policies formulated by the decision makers and policy planner’s extensive analysis on student and employee related data. Over all 80 to 85% of decisions are made based on the reports generated by the proposed system. The realistic productivity is about 85%.

V. FUTURE SCOPE

The enhancement that can be carried out on the present system is the implementation of the real time ETL system. Real time ETL refers to the software that moves data synchronously into a data warehouse with some urgency-within minutes of the execution of the business transaction. Implementation of real-time data warehouse reflects a new generation of hardware, software and techniques. Capture, Transform, and Flow (CTF) is a relatively new category of data integration tools designed to simplify the movement of real-time data across heterogeneous database technologies. The transformation functionality of CTF tools is typically basic in comparison with today’s mature ETL tools, so often real time data warehouse CTF solutions involve moving data from the operational environment, lightly transforming it using the CTF tool and then staging it.

REFRENCES

[1] Ralph Kimball, the Data Warehouse ETL Toolkit, *Wiley India Pvt Ltd.*,2006.

[2] KV. K. K. Prasad, Data warehouse development Tools, *Dreamtech Press*, 2006.

[3] W. H. Inmon, Building the Data Warehouse. *Wiley; 3rd edition*.

[4] Alex Berson, Data Warehousing Data Mining & OLAP, Computing *Mcgraw-Hill,* November 5, 1997.

[5] Arshad Khan, SAP and BW Data Warehousing, *Khan Consulting and Publishing, LLC* (January 1, 2005)

[6] Carlo DELL’AQUILA,‘An Academic Data Warehouse’ *World Scientific and Engineering Academy and Society (WSEAS) Stevens Point*, Wisconsin, USA ©2007.