Data Warehousing: Concepts and Mechanisms

Mukesh Yadav

Department of Computer Sc., GITM, Gurgaon
Email: yadavm@outlook.com

Abstract : In the last years, data warehousing has become very popular in organizations. The size of the data warehouse market is expected to be at least $8 billion at the end of 1998, and more than 900 vendors provide various kinds of hardware, software, and services for data warehousing. The research community noticed this trend as well and determined data warehousing as one of the "hot topics". In this paper, we introduce the basic concepts and mechanisms of data warehousing.

The Aim of Data Warehousing

Data warehousing technology comprises a set of new concepts and tools which support the knowledge worker (executive, manager, analyst) with information material for decision making. The fundamental reason for building a data warehouse is to improve the quality of information in the organization. The key issue is the provision of access to a company-wide view of data whenever it resides. Data coming from internal and external sources, existing in a variety of forms from traditional structural data to unstructured data like text files or multimedia is cleaned and integrated into a single repository. A data warehouse (DWH) is the consistent store of this data which is made available to end users in a way they can understand and use in a business context.

The need for data warehousing originated in the mid-to-late 1980s with the fundamental recognition that information systems must be distinguished into operational and informational systems [5]. Operational systems support the day-to-day conduct of the business, and are optimized for fast response time of predefined transactions, with a focus on update transactions. Operational data is a current and real-time representation of the business state.

In contrast, informational systems are used to manage and control the business. They support the analysis of data for decision making about how the enterprise will operate now and in the future. They are designed mainly for ad hoc, complex and mostly read-only queries over data obtained from a variety of sources. Informational data is historical, i.e., it represents a stable view of the business over a period of time. Limitations of current technology to bring together information from many disparate systems hinder the development of informational systems. Data warehousing technology aims at providing a solution for these problems.

The Main Characteristics of Data Warehouse Data

Data in the DWH is integrated from various, heterogeneous operational systems (like database systems, flat files, etc.) and further external data sources (like demographic and statistical databases, WWW, etc.). Before the integration, structural and semantic differences have to be reconciled, i.e., data have to be "homogenized" according to a uniform data model. Furthermore, data values from operational systems have to be cleaned in order to get correct data into the data warehouse.

The need to access historical data (i.e., histories of warehouse data over a prolonged period of time) is one of the primary incentives for adopting the data warehouse approach. Historical data are necessary for business trend analysis, which can be expressed in terms of understanding the differences between several views of the real-time data (e.g., profitability at the end of each month). Maintaining historical data means that periodical snapshots of the corresponding operational data are propagated and stored in the warehouse without overriding previous warehouse states. However, the potential volume of historical data and the associated storage costs must always be considered in relation to their potential business benefits.

Furthermore, warehouse data is mostly non-volatile, i.e., access to the DWH is typically read-oriented. Modifications of the warehouse data take place only when modifications of the source data are propagated into the warehouse.

Finally, a data warehouse contains usually additional data, not explicitly stored in the operational sources, but derived through some process from operational data (called also derived data). For example, operational sales data could be stored in several aggregation levels (weekly, monthly, quarterly sales) in the warehouse.

Data Warehouse Systems

A data warehouse system (DWS) comprises the data warehouse and all components used for building, accessing and maintaining the DWH (illustrated in Figure 1). The center of a data warehouse system is the data warehouse itself. The data import and preparation component is responsible for data acquisition. It includes all programs, applications and legacy systems interfaces that are responsible for extracting data from operational sources, preparing and loading it into the

warehouse. The access component includes all different applications (OLAP or data mining applications) that make use of the information stored in the warehouse.

Additionally, a metadata management component (not shown in Figure 1) is responsible for the management, definition and access of all different types of metadata. In general, metadata is defined as "data about data" or "data describing the meaning of data". In data warehousing, there are various types of metadata, e.g., information about the operational sources, the structure and semantics of the DWH data, the tasks performed during the construction, the maintenance and access of a DWH, etc. The need for metadata is well known. Statements like "A data warehouse without adequate metadata is like a filing cabinet stuffed with papers, but without any folders or labels" characterize the situation. Thus, the quality of metadata and the resulting quality of information gained using a data warehouse solution are tightly linked.

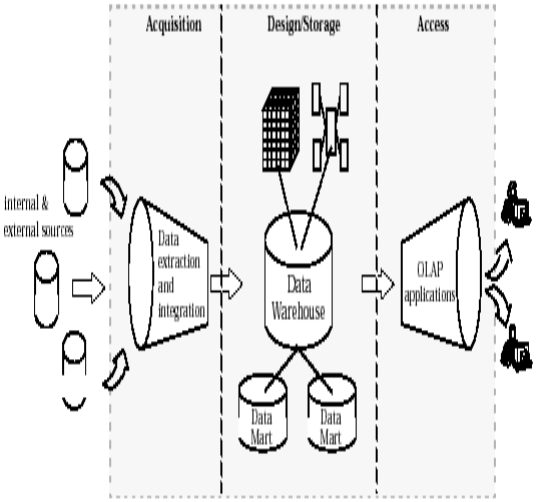

Figure 1 A typical data warehouse system architecture

Implementing a concrete DWS is a complex task comprising two major phases. In the DWS configuration phase, a conceptual view of the warehouse is first specified according to user requirements (data warehouse design). Then, the involved data sources and the way data will be extracted and loaded into the warehouse (data acquisition) is determined. Finally, decisions about persistent storage of the warehouse using database technology and the various ways data will be accessed during analysis are made.

After the initial load (the first load of the DWH according to the DWH configuration), during the DWS operation phase, warehouse data must be regularly refreshed, i.e., modifications of operational data since the last DWH refreshment must be propagated into the warehouse such that data stored in the DWH reflect the state of the underlying operational systems. Besides DWH refreshment, DWS operation includes further tasks like archiving and purging of DWH data or DWH monitoring.

Data Warehouse Design

Data warehouse design methods consider the read-oriented character of warehouse data and enable the efficient query processing over huge amounts of data. A special type of relational database schemas, called star schema, is often used to model the multiple dimensions of warehouse data (in contrast to the two-dimensional representation of normal relational schemas). In this case, the database consists of a central fact table and several dimension tables. The fact table contains tuples that represent business facts (measures) to be analyzed, e.g., sales or shipments. Each fact table tuple references multiple dimensional table tuples each one representing a dimension of interest like products, customers, time, region or salesperson. Dimensions usually have associated with them hierarchies that specify aggregation levels and hence granularity of viewing data (e.g., day-> month -> quarter ->year is a hierarchy on the time dimension [1]). Since dimension tables are not normalized, joining the fact table with the dimension tables provides different views (dimensions) of the warehouse data in an efficient way. A variant of the star schema, called snowflake schema, is commonly used to explicitly represent the dimensional hierarchies by normalizing the dimension tables.

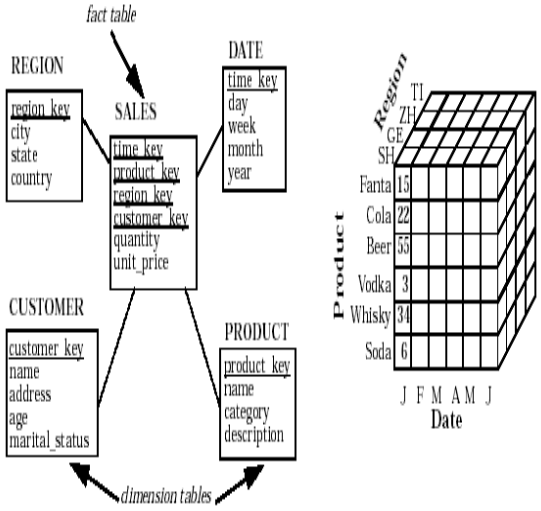

Figure 2 Star schema and data cube

A more natural way to consider multidimensionality of warehouse data is provided by the multidimensional data model. Thereby, the data cube is the basic underlying modeling construct. Special operations like pivoting (rotate the cube), slicing-dicing (select a subset of the cube), roll-up and drill-down (increasing and decreasing the level of aggregation) have been proposed in this context. For the implementation of multidimensional databases, there are two main approaches. In the first approach, extended relational DBMSs, called relational

OLAP (ROLAP) servers, use a relational database to implement the multidimensional model and operations. ROLAP servers provide SQL extensions and translate data cube operations to relational queries. In the second approach, multidimensional OLAP (MOLAP) servers store multidimensional data in non-relational specialized storage structures. These systems usually precompute the results of complex operations (during storage structure building) in order to increase performance.

Data Acquisition

For the first task of data acquisition, the data extraction, standard interfaces (e.g., ODBC, EDA/SQL) or gateways are often used in commercial extraction tools and proprietary extraction scripts. Although often underestimated, data extraction is one of the most time-consuming tasks of data warehouse development, specially when older legacy systems must be integrated.

Usually, data extracted from operational systems contains lots of errors, and must be first transformed and cleaned before loading it into the data warehouse. Data values from operational systems can be incorrect, inconsistent, unreadable or incomplete. Furthermore, different formats and representations may be used in the various operational systems. Particularly, for the integration of external data, data cleaning is an essential task in order to get correct and qualitative data into the data warehouse, and includes the following tasks:

• convert data to the common, internal warehouse format from a variety of external representations

• identify and eliminate duplicates and irrelevant data

• transform and enrich data to correct values (e.g., by checking the membership of an attribute in a list)

• reconcile differences between multiple sources, due to the use of homonyms (same name for different things), synonyms (different names for same things) or different units of measurement.

After cleaning, data that comes from different sources and will be stored in the same warehouse table must be merged and possibly set into a common level of detail. Furthermore, time-related information (update/extraction/load date) is usually added to the warehouse data to allow the construction of histories. As mentioned above, one of the main characteristics of a data warehouse is the creation and storage of new base data (compared to the contents of operational systems). Thus, beyond extracting and integrating existing operational data, derived and aggregated data must be calculated using appropriate functions or rules. Finally, before or during loading data into the warehouse, further tasks like filtering, sorting, partitioning and indexing are often required [14]. Populating the target warehouse is then performed using a DBMS's bulk data loader or an application with embedded SQL.

Data Storage and Access

The special nature of warehouse data and access necessitates adjusted mechanisms for data storage, query processing and transaction management. Complex queries and operations involving large volumes of data require special access methods, storage structures and query processing techniques. For example, bitmap indices and various forms of join indices (e.g., Starjoin [10], parallel join [9]) can be used to significantly reduce access time. Furthermore, since access to warehouse data is mostly read-oriented, complex concurrency control mechanisms and transaction management must be adapted [14]. Settling subsets of it in form of data marts can also speed up access to the data warehouse. A data mart is a selected part of the data warehouse, which supports specific decision support application requirements of a company's department or geographical region. It usually contains simple replicates of warehouse partitions or data that has been further summarized or derived from base warehouse data. Instead of running ad hoc queries against a huge data warehouse, data marts allow the efficient execution of predicted queries over a significantly smaller database.

Virtual Data Warehouses

The proposal of virtual data warehouses is considered as a way to rapidly implement a data warehouse without the need to store and maintain multiple copies of the source data. Virtual data warehouses often provide a starting point for organizations to learn what end users are really looking for. End-users have the possibility to directly access real-time source data using advanced networking capabilities tools. The drawbacks of this approach compared to the classical data warehouse approach illustrated in Figure 1 are:

• data quality and consistency is not guaranteed since no prior data preparation (reconciliation) takes place,

• historical data is usually not available,

• end-user access time is usually unpredictable depending on the availability of operational sources, network load (which is high in this approach), query complexity and translations between different database formats.

State-of-the-art

Today, a plethora of tools, particularly for specific tasks of a DWS like data acquisition, access and management is available in the market. For the implementation of a complete DWS, a set of tools must be integrated to form a concrete warehousing solution. The ultimate integration goal is to avoid interface problems. The trend is towards "open"-solutions (supported e.g., by IBM Visual Warehouse [8], HP Open Warehouse [7] or Prism Solutions [11]), which gives the opportunity to combine several tools in one DWS. For example, the HP Open Warehouse is a framework for designing data

warehouses based on HP- and third party hardware and software components. HP-customers can choose from solutions in areas such data extraction and transformation, relational databases, data access and reporting, OLAP, web-browsers applications and data mining.

A further recent market trend is the adoption of data marts as a way to use and experiment with data warehouse technology in particular departments (e.g., marketing). Linking data warehouse to the Internet (as additionally data source or access interface) gains more attention because it allows companies to extend the scope of warehouse to external information. Until now, the research community attempts to solve particular problems, mostly using well-known concepts and research results from other research fields (like materialized views, index selection, data partitioning) [2, 14]. The most prominent research project, the WHIPS project at the University of Stanford, investigates a wide spectrum of data warehousing problems based on techniques of materialized views [13]. In Switzerland, the "Kompetenzzentrum Data Warehousing Strategie" (CC DWS) at the University of St. Gallen (HSG) focuses, together with a number of companies, at the development of a process model for the successful introduction of data warehousing in big companies. Our work in the context of the SIRIUS project focuses on the investigation of techniques for the incremental refresh [6]. In the SMART project (a cooperation with Rentenanstalt/Swiss Life), we investigate the design and implementation of a metadata management system for a data warehouse environment. Developing a data warehouse system is an exceedingly demanding and costly activity, with the typical warehouse costing in excess of $1 million [12]. Nevertheless, data warehousing has become a popular activity in information systems development and management.

According to the market research firm Meta Group, the proportion of companies implementing data warehouses exploded from 10% in 1993 to 90% in 1994, and the data warehousing market will expand from $2 billion in 1995 to $8 billion in 1998. Improving access to information and delivering better and more accurate information, is for more and more companies a motivation for using data warehouse technology.

REFERENCES

[1] R. Agrawal, A. Gupta , S. Sarawagi. Modeling Multidimensional Databases. Proc. Of the 13th Intl Conference on Data Engineering, Birmingham U.K., April 1997.

[2] S. Chaudhuri, U. Dayal. An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record, 26:1, March 1997.

[3] Digital Consulting Inc. Data Warehouse Trends '98. White Paper available from http://www.dw-institute.com/pubsindex.htm.

[4] M. P. Burwen. Database Solutions. White Paper available from http://www.dw-institute.com/pubsindex.htm.

[5] B. Devlin. Data Warehouse from Architecture to Implementation. Addison-Wesley, 1997.

[6] S. Gatziu, A. Vavouras, K. R. Dittrich. SIRIUS: An Approach for Data Warehouse Refreshment. Technical Report 98.07, Department of Computer Science, University of Zurich, June 1998.

[7] Hewlett Packard Company. http://www.hp.com/esy/solutions/data_warehousing.

[8] IBM Corporation. http://www.software.ibm.com/data/vw/.

[9] C. Lee, Z.A. Chang. Utilizing Page-Level Join Index for Optimization in Parallel Join Execution. IEEE Transactions on Knowledge and Data Engineering, 7(6), December 1995.

[10] O'Neil, G. Graefe. Multi-Table Joins through Bitmapped Join Indices. SIGMOD Record, 24(3), September 1995.

[11] Prism Solutions. http://www.prismsolutions.com.

[12] H. J. Watson, B. J. Haley. Data Warahousing: Managerial Considerations. Communications of the ACM, 41(9), September 1998.

[13] J. Wiener, H. Gupta, W. Labio, Y. Zhuge, H. Garcia-Molina, J. Widom. A System Prototype for Warehouse View Maintenace. Proc. of the ACM Workshop on Materialized Views: Techniques and Applications, Montreal, June 7, 1996.

[14] M-C. Wu, A. P. Buchmann. Research Issues in Data Warehousing. Datenbanksysteme in Büro, Technik und Wissenschaft: GI-Fachtagung, Springer-Verlag, Ulm, 1997.