QUERY PROCESSING AND ANALYSIS FROM DISTRIBUTED DATA WAREHOUSES

Khine Ohnmar Htun, Win Lai Lai Phyu
University of Computer Studies, Yangon
khinelay20@gmail.com, winlai@gmail.com

ABSTRACT

Data warehouses must have efficient Online Analysis Processing (OLAP) that provides tools to satisfy the information needs of business managers, helping then to make faster and more effective decisions. The increasing use of decision support systems led to an explosion in the amount of business information that must be managed by the data warehouse. Improving query processing and analysis in such an environment is very difficult and can only be achieved by a combination of different approaches, in particular the use of materialized views, distributed data warehouses. This system tailored design framework for distributed data warehouse and OLAP. In this paper, we process and analyze the performance of distributed data warehouse with typical OLAP operations using different types of queries and this system can produce a variety of reports and information in time with accurately fast response for performance evaluation and decision making.

1. INTRODUCTION

Effective decision-making is vital in a global competitive environment where business intelligence systems are becoming an essential part of virtually every organization. The core of such systems is a data warehouse, whose stored historical and consolidated data from the transactional databases, supporting complicated ad-hoc queries that reveal interesting information. The so-called On-Line Analytical Processing (OLAP) queries typically involve large amounts data and their processing should be efficient enough to allow interactive usage of the system. These systems assume that the users belong to the organization that owns the data warehouse and have access to the proprietary infrastructure. The query requirements are well defined and the problems are related to data placement, materialized view selection and query optimization, given a static network of servers.

We focus on OLAP for several reasons: (i) OLAP data have a regular structure which allows easy decomposition and reuse of previous results, (ii) the size of the results is typically large and justifies the overhead of searching neighbor clients, (iii) updates in data warehouses are infrequent compared to transactional databases, therefore the cached data are valid for a long time and (iv) queries exhibit temporal and geographically location.

The rest of the paper is organized as follows: in Section 2, we include some essential review the related work. Section 3 presents an overview of the OLAP architecture with distributed data warehouses in multidimensional data model dimensional model for employee payroll system and schema for multidimensional databases while in Section 4 we present the experimental results for the variety of the reports. Finally, Section 5 concludes the paper with a discussion.

2. RELATED WORK

Data warehousing is architecture to help business executives to understand and organize data and make a business decision. A data warehouse is a subject oriented, integrated, time variant, non volatile collection of data in support of decision making processes [2]. Data warehousing approach is to integrate information and heterogeneous sources in advance, store the historical information in a warehouse and support complex multidimensional queries. On-Line Analytical Processing (OLAP) manages data warehouses for data analysis and provides calculations such as summarization and aggregation in advance, and manages information at different levels of granularity. OLAP has become very popular techniques to help users analyze data by providing multiple views of the data. Data warehouses and OLAP tools are based on a multidimensional data model. Information is derived from On-Line Analytical Processing (OLAP) systems used for analysis, planning and management reporting through access to a variety of sources. An OLAP system usually references information that is stored in a data warehouse. Use of this technology provides the facility to present a comprehensive view of the State enterprise.

The most common strategies to improve query processing and analysis in warehousing environments are the pre-computation of data in the form of materialized views and the use of special indexes structures. However, OLAP queries are most frequently of an ad hoc nature, which restricts the pre-computation to the imagination of the distributed data warehouse. Users might query on dimensions that are not materialized in views. The indexing structures provide faster access to the data stored in the warehouse, but increase the size of the tables stored in that data warehouse. One can say that the strategies presented provide faster query processing, but they may not be fast enough as perceived by the user. Distributed processing techniques have been applied to relational database systems. The basic idea behind distributed databases is to carry out evaluation steps in parallel whenever possible, in order to improve performance. Although some vendors support distributed data warehousing to various degrees, the fact is that distributed query processing in data warehouses has received little attention in research community. [6]

In DWS, typical OLAP queries are transformed into *N* partial queries that are executed in distributed in each of *N* computers that represent the DWS system, avoiding the need of communication between computers during query execution. The distributed execution of a query by the available computers maintains the best load balance because the number of fact rows stored in each computer is about the same and the row distribution is random. The query processing and analysis is a DWS layer responsible to receive the original query from the client application, rewriting it if necessary, and distributed this "modified" query by all computers.

3. OLAP ARCHITECTURE WITH DISTRIBUTED DATA WAREHOUSES

The warehouse may be distributed for load balancing, higher performance, and higher availability. In such a distributed architecture, the metadata repository is usually replicated with each fragment of the warehouse, and the entire warehouse is administered centrally. An alternative architecture, implemented for expediency when it may be too expensive to construct a single logically integrated enterprise warehouse, is a federation of warehouses or data marts, each with its own repository and decentralized administration. Designing and rolling out a data warehouse is a complex process, consisting of the activities. Define the architecture, do capacity planning, and select the storage servers, database and OLAP servers, and tools Integrate the servers, storage, and client tools. Design the warehouse schema and views. Define the physical warehouse organization, data placement, partitioning, and access methods [5].

In addition, however, there are certain logical relations and OLAP operations that are much easier to perform in multidimensional databases. These operations include: (i) defining parent-child

relations between dimensions and constructing dimensional hierarchies across geography, organization, time and other important organizing concepts; (ii) easily performing matrix calculations that allow whole vectors or slices of arrays to be operated on at once; (iii) ranging or sub-setting (also known as "dicing") multidimensional arrays to provide more focused descriptions, reports and analyses; (iv) rotation (also known as "data slicing") to examine a different view of the multidimensional array being queried without having to reassemble the array from basic data; and (v) aggregating or disaggregating multidimensional arrays to display higher or lower levels in a dimensional hierarchy such as time period, geography, or organization (known as "rolling-up" or "drilling-down").

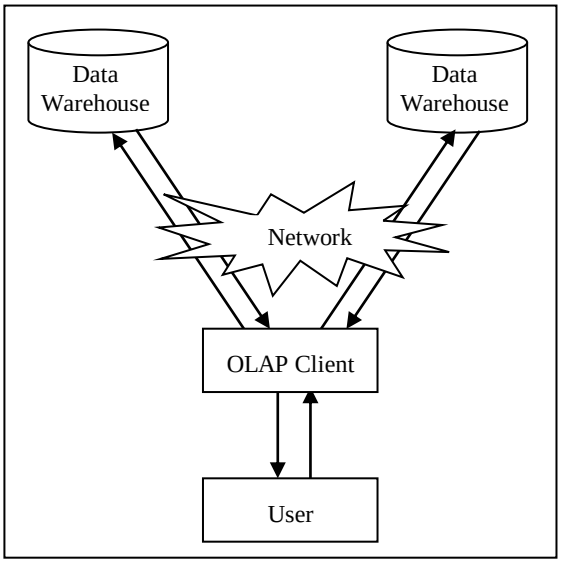


Figure 1. System design OLAP architecture with distributed data warehouses

This system includes tools for extracting data from multiple operational databases and external sources; for cleaning, transforming and integrating these data; for loading data into the data warehouse; and for periodically refreshing the warehouse to reflect updates at the sources and to remove data from the warehouse, perhaps onto slower archival storage. Most of the queries over a star schema can be transformed into *N* independent partial queries. In reality, query processing and analysis in OLAP client could need to access intermediate results computed from distributed data warehouses in Servers: (Head Office Data Source and Branch Data Source). Thus, OLAP architecture concerning the need of communication among Servers during query execution needs communication among servers to compute the final results. In these queries, the computing of the partial results in each server also needs to access data stored in data warehouse. In this case, the execution of queries is dependent of partial results which are calculated over the data in data warehouse system of the servers as shown in Figure 1.

3.1. OLAP Operations in Multidimensional Data Model

In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies [3]. OLAP is the storage of multidimensional, generally hierarchical, data providing near constant-time answers to quires. OLAP include roll-up, drill down, slice and dice and pivot as shown in Figure 2.

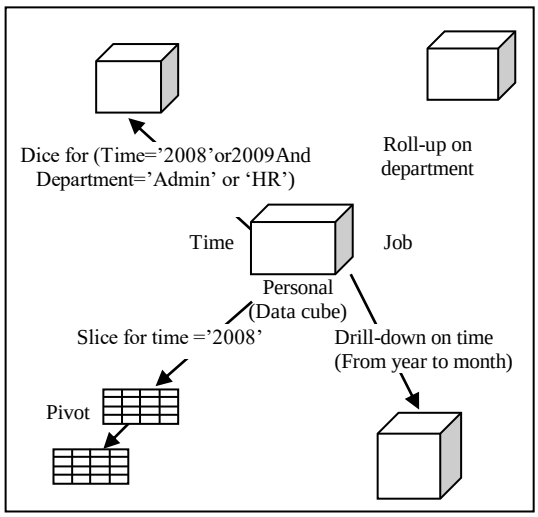


Figure 2. Example of typical OLAP operations in multidimensional data

Roll-Up: Roll-up operation performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension reduction. When roll-up operation is performing by dimension reduction, one or more dimension is removing from the given cube.

Drill-Down: Drill-down is the reverse of roll-up. It navigates from less detail data to more detail data. Drill-down can be realized by their stepping down concept hierarchy for a dimension or introducing additional dimensions.

Slicing and Dicing: Selecting a subsection of data cube based on the constants in one or few dimensions. If one dimension is fixed, the operation is called slice and if more than one dimension are fixed, the operation is called dice.

Pivot: Pivoting is swapping of columns and rows. This allow user to look at data from different view. This is also commonly known as rotation.

3.2. Dimensional Model for Employee Payroll System

The fundamental idea of dimensional modeling is that nearly every type of business data can be represented as a kind of cube data, where the cells of the cube contain measured values and the edges of the cube define the natural dimensions of the data. [7]

Facts: Fact table represents a relationship between two or more dimension entities and has some measurement attributes. Fact table contain two types of columns that is facts and foreign keys to dimension tables. Salary is fact and Payroll_ID, P_ID, Job_ID, Time_ID, and Leave_ID are foreign keys.

Dimensions: A dimension is a collection of text-like attributes that are highly correlated with each other. The dimensions are syntactical categories that all allow us to specify multiple ways to look at business information, according to natural perspectives under which its analysis can be performed. Each dimension can be organized into a hierarchy of levels, corresponding to data domains at different granularity. A level can have description associated with it. Within a dimension, values of different levels are related through a family of rollup functions.

Variable: Variables are typical numerical measures like Department, Position, Salary for each employee and Total Net Pay of each Department.

Hierarchy: A hierarchy is path of aggregation of dimensions. A dimension may have multiple levels of granularity, which have parent-child relationship. A hierarchy defines how these levels are related.

Member: A member is a data item in the dimensions. Typically, we create a caption or describe a measure of database using members.

OLAP Queries: OLAP queries are sets of pre computed views or special data structure (e.g. multidimensional arrays) of selected data that are formed by aggregating values across attributes combinations (a group in the database terminology).

3.3 Schema for Multidimensional Databases

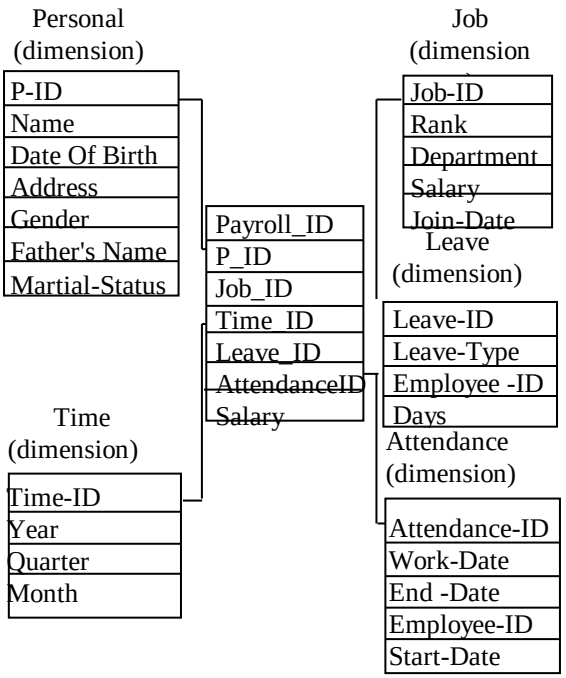
The entity-relationship data model is commonly used in design of relational databases, where a database schema consists of a set of entities and the relationships between them. Such a data model for a data model is appropriate for on-line transaction processing. A data warehouse, however, requires a concise, subject-oriented schema that facilitates on-line 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 snow-flake schema, or fact constellation schema. This system used the star schema in Table 1.

This model consists of a set of dimensions and a set of measures (facts), where each dimension is typically arranged in a hierarchy. All dimensions have some attributes, describing a different perspective for the analysis of business. For instance, in the classical of a chain of stores business, the dimensions are Personal, Attendance, Job, Time and Leave. Each cell within the multidimensional structure contains measures (typical numerical facts) along each of the

dimensions. A single cell may contain the salary for a given employee in an employee payroll data in a single month. The conceptual multidimensional data model can be physically realized in two days, (i) by using trusted relational databases (star schema) or (ii) by making use of specialized multidimensional databases.

In this paper, we assume that a multidimensional database is a relational data warehouse in which the information is organized as a star schema. It offers flexibility, but often at the cost of performance because more joins for each query are required. A star schema models a consistent set of facts (aggregated) in a central fact table and the descriptive attributes about the facts are stored in multiple dimension tables. The equivalent star schema of the example of a chain of stores presented before is shown in Table 1. The fact table is Salary and the dimensions are Personal, Attendance, Job, Time and Leave. The size of fact table is equal or much bigger than the size of dimension tables. The size of dimension tables could be hundreds or thousands of rows while the fact table could be millions of rows [7]. Queries tend to use aggregations and join on two or more tables, have often a large computation time, and are mainly ad hoc in nature.

Table 1. Star schema



Using DWS, the number of fact rows stored in each server is about the same, meaning that each server has a fraction of the fact rows. The data warehouse contains a large central table (Fact table) containing the bulk of data, with no redundancy, and a set of smaller attendant tables (dimension tables), one for each dimension. These Schema graph in Table 1 resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.

4. EXPERIMENTAL RESULTS

A common methodology to construct data warehouses is to start with some local data marts (e.g., one data mart for each department). The knowledge acquired during this phase can be used to construct in parallel a global enterprise schema for the data warehouse. The requirements of the analysts will grow in time, and after some time they want to make queries across several data marts of the departments. At this point, the enterprise wide data warehouse comes into play: it can either be a virtual/distributed data warehouse,

This system use C# programming language and SQL server 2005 for query processing and analysis from distributed data warehouse. In this system, the user can view the Employee Payroll information from the system. The main form includes the three process operations; SETUP, DATA and QUERY process. In this system, the user makes the server setup and tests the server connection. There are two Data Sources in the system; Head Office Data Source and Branch Data Source. IP Address, DB server and Databae can enter the input text boxes each data source and save these data.

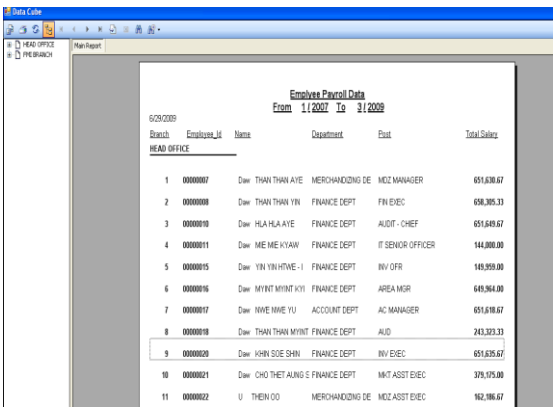


Figure 3. Show payroll data for all branch

Figure 4. Rollup operation for total salary

Figure 5. Drill down operation for the total salary

Figure 6. Dice view for salary information

The user also loads the payroll data from Brach Data Source. After loading the branch payroll data, the system produces the results of all payroll information reports in Figure 3 and it also performs OLAP queries: rollup for total salary of each year and desire location as shown Figure 4, drill down for the total salary information from each year, desire location and desire department as shown in Figure 5and dice view for salary information in financial year, location, department, all employees, and employee detail salary in desired location, desire department and minimum/maximum salary as shown in Figure6.

5. CONCLUSION

This system is to develop data mining process for user required and /or interested query or analyze or report concerning with payroll, leave records. This will be a useful tool to analyze the payroll condition for stationery item and/or stationery category and each branch of the payroll. This system implements the reports in the Payroll conditions of each department and branch conditions of each employee. This system supports distributed query processing and analysis from distributed data warehouse.

REFERENCES

[1] L. Kerschberg, "Knowledge Management in Heterogeneous Data Warehousing Environments", Co-Director, E-Center for E-Business, Department of Information and Software Engineering, George Mason University, MSN 4A4, 4400 University Drive, Fairfax, VA 22030-4444, USA.

[2] K. Strange, "The Challenges of Implementing a Data Warehouse to Achieve Business Agility", Colorado Convention Center, Denver, Colorado, 7-10 May 2001

[3] S. Azhar, S. M. Ahmed, I. Ahmad, "Safety Information Management in Construction Firms: A Data Warehousing Approach." Department of

Building Science, Auburn University, Alabama, USA.

[4] K. D. Schewe, J. Zhao, "Balancing Redundancy and Query Costs in Distributed Data Warehouses", Massery University, Information Science Research Center, Department of Information Systems, Palmerston North, New Zealand.

[5] R. Sharma, K. Shah, " Data Warehouse and OLAP Technology", part-1, Group no. 3.

[6] A. Sen, A. P. Sinha, " A Comparison of Data Warehousing Methodologies".

[7] S. Chaudhuri, U, Dayal, "An Overview of Data Warehousing and OLAP Technology", ACM Sigmod Record, March 1997.