Technology Challenges in a Data Warehouse

Ramesh Bhashyam

NCR Corporation
Teradata Research and Development
San Diego, California
USA
Bhashyam.ramesh@ncr.com

Abstract

This presentation will discuss several database technology challenges that are faced when building a data warehouse. It will touch on the challenges posed by high capacity drives and the mechanisms in Teradata DBMS to address that. It will consider the features and capabilities required of a database in a mixed application environment of a warehouse and some solutions to address that.

A data warehouse integrates large volumes of detailed and current data across entire organizations and enables different forms of decision making from the same data base. It provides a unified view of operational and historical data. It will often use as a foundation a detailed data model and enable different summaries or views, as appropriate to the business, to be built on top of the detailed model.

The usage of a data warehouse has evolved from reporting and decision support system to mission critical decision-making operational systems. Data warehouses are often used for mining types of applications. These applications read massive volumes of data from within the data warehouse and are demanding of both CPU and IO resources. Data warehouses can also be used for operational decision-making applications, applications which read a small but well-focused set of data from the warehouse, and use few resources. These operational applications are differentiated by short response time requirements, making scalability a challenge. Data warehouses that combine both these types of applications require that the operational data be integrated, current and up to date. The common belief that warehouse data is static is no longer valid.

An emerging requirement is that warehouses should have the ability to detect events and enable actions based on complex analysis. For example, the decision to replenish an item may be based on wider criteria than simply low inventory levels.

In this talk we will examine some of the technology challenges to a successful implementation of a data warehouse.
Various components of a data warehouse hardware platform have seen impressive improvements but these various improvements have not been equally powerful. This has led to imbalances in comparative performances. For example, although the CPU power has followed Amdahl's law and has improved significantly, the ability to supply the processor with data has not improved to the same degree. This imbalance is best observed in the disk subsystem. Disk storage capacities have gone up from few GB a few years ago to hundreds or even thousands of GB now - a factor of several hundred. However, the access rates have gone up by only a factor of three or four. So the challenge is how to adopt the latest high capacity drives while not impacting performance for IO intensive applications. Some of the techniques that Teradata uses in this context will be detailed.

It is well accepted that the only viable solution for large warehouses is a shared nothing MPP platform. But there are challenges to making operational queries demonstrate scalable performance on such platforms. While the optimizer, combined with proper data distribution, are important prerequisites for making complex queries scale, more is needed from the execution engine and the optimizer for making very short queries similarly

**Proceedings of the 30th VLDB Conference,
Toronto, Canada, 2004**

scalable. This must be achieved without affecting complex query scalability. Typical solution such as table duplication and storing tables in few nodes are inappropriate for complex queries although they are good for operational queries.

An important area for operational query performance is indexes. Operational queries access a focused set of data. They may access few rows from a number of tables and join them. These queries are not resource intensive but require short response times. Different kinds of secondary indexes that enrich the mechanisms available for accessing base table rows are therefore important. These include both local indexes and global indexes. In addition various forms of materialized views such as Teradata's Join Indexes and aggregate join indexes are important. The join indexes are also appropriate for complex queries besides short operational queries.

Managing a multi-user system with very different application profiles requires sophisticated workload management. Workload management implies maximizing system throughput while meeting widely varying service levels. Management can be broadly classified under two distinct categories - controlling access to the system and managing access once inside the system. While both of these are non-trivial problems, they are even more challenging in a parallel architecture where multiple threads and processes execute on behalf of a query from within a single node and from across multiple nodes.

Finally, integrating a warehouse to the external system and closing the loop by taking action based on analysis is the most recent technology challenge. Most existing technologies allow interaction with a data warehouse using a pull model. In such a model the application or user periodically polls the database for analysis or state changes. However a push model becomes essential for data warehouses that must integrate and provide notifications on actionable events to external systems. In a push model a change in "status" is evaluated and a user or application is asynchronously notified. It is also important to understand what a change means. The definition of a change must evolve from simple state changes such as "inventory below a level" to complex analytics based state changes. Such an environment requires that data ingest be asynchronous with data processing which must be asynchronous with analysis output through events. Simple database triggers therefore are insufficient for this model.

This talk will address some or all of these warehouse technology challenges.