2000

Data Warehouse Benchmarking

Frank Graziani
University of North Carolina at Greensboro, frankgraziani@mindspring.com

Hamid Nemati
University of North Carolina at Greensboro, nemati@uncg.edu

Glen Piper
University of North Carolina at Greensboro, piper@wfu.edu

Steve Rose
University of North Carolina at Greensboro, steven.rose@wachovia.com

Data Warehouse Benchmarking

Hamid Nemati, The University of North Carolina at Greensboro, ISOM Department, nemati@uncg.edu
Frank Graziani, The University of North Carolina at Greensboro, ISOM Department, frankgraziani@mindspring.com
Glen Piper, The University of North Carolina at Greensboro, ISOM Department, piper@wfu.edu
Steve Rose, The University of North Carolina at Greensboro, ISOM Department, Steven.Rose@wachovia.com

Abstract

This paper is concerned with the subject of performance measurement and benchmarking in Data Warehouses. We aim to clarify the issues surrounding the concept of performance measurement in Data Warehouses by examining and discussing the general themes of benchmarking, performance, and current industry standards. Data Warehouse performance measurement into two areas: objective measures and subjective measures. We conclude by providing specific guidelines for benchmarking a data warehouse project.

Introduction

A Data Warehouse is an information architecture designed to support the strategic decision making activities of an organization in a fashion that is cannot be achieved with traditional operational and legacy systems (Berson and Smith, 1997). It is not a project with an end (Inmon 1996). In fact, it is an on-going project that requires constant tuning, adjustments, and upgrading. Data Warehouse performance measurement and benchmarking play important roles in the on-going management of a Data Warehouse. Data Warehouses are unlike On-line Transaction Processing (OLTP) systems and have performance problems all to there own. If these problems are not addressed promptly and properly, they can lead to the failure of the Data Warehouse. Usually, performance problems will not occur all at one time, but rather start small and develop into massive problems as the Data Warehouse grows. Catching and resolving a problem as quickly will save time, money, effort, and perhaps most importantly keep the problems transparent to the users. The importance of measuring a Data Warehouse's effectiveness is easily understood. Simply, Data Warehouses are justified on the basis of the information they make available to the decision-makers. Without any measures it is impossible to determine whether the warehouse has added value. There are two basic reasons to conduct performance measurement of a Data Warehouse: to measure its health and to measure its effectiveness.

Like Data Warehousing, benchmarking is in its infancy. Industry standards are still being developed and will be for many years to come. Benchmarking in the Computer Industry as a whole has had a rather colorful history. In fact, it is often referred to as "benchmarketing." This refers to the practice of aggressively using benchmarks as a way of differentiating one's product from one's competitors, possibly unfairly. This, unfortunately, led to abuses where a company might try to take unfair advantage by adding special tuning to a system during a test, or might want to emphasize one particular set of results while downplaying others. These practices occurred throughout the computer industry, not just in data warehousing, but have left an overall mistrust of published benchmarks. So-called "Benchmark wars" may start after someone publishes benchmark results. Competitors attempt to bring in specialists and try to get new and winning numbers. The original company will then attempt to get better number using their experts. This often continues for several iterations. Special software changes may be applied, with the promise that they will be used in later versions of the program. Even valid benchmark results can be used in deceptive ways. For example, ratios can be dangerous if not viewed in context. The difference between a change in execution time from ten seconds to five may not be true if the experiment is repeated 1000 or one million times, so 2:1 or "twice as fast" may only be true in certain limited cased.

Data Warehouse Benchmarks

Given the current state of Data Warehousing industry and the importance of benchmarking, there have been a number of attempts to establish benchmarking standards. Among these, the following are noteworthy. *The University of Wisconsin benchmark,* was developed as an attempt to provide a viable, third party alternative to vendors performing tests using their own benchmarks. ***A****NSI* ***S****QL* ***S****tandard* ***S****calable and* ***P****ortable* ***(AS3AP)*** benchmark determines an equivalent database size, which is the maximum size of the AS3AP database for which the system is able to perform the designated AS3AP set of single and multi-user tests in under 12 hours. *The Datamation benchmark* is an old industry standard that requires the entrants to sort 1 million 100-byte records. The result is measured in seconds. *SAP R/3 Sales and Distribution (SD) benchmark* is a benchmark available to hardware vendors who want to demonstrate performance with this one package. It is design to simulate typical usage in a business environment. The results of the test are measured in SAP's. *Red Brick's Proof of Performance & Scalability* is a benchmark that Red Brick published to showcase its Data Warehousing products. It is designed to demonstrate performance and scalability in both loading data and running sophisticated queries. However, the most widely used data warehouse benchmarks are those provided by the Transaction Processing Performance Council (TPC). TPC is the foremost authority in the arena of data warehouse and OLTP system performance measurement and benchmarking. The TPC is a non-profit consortium founded to define transaction processing and database benchmarks, as well as to disseminate objective, verifiable TPC performance data to the industry

Factors Affecting the Performance of a Data Warehouse

There are a number of factors affecting the performance of a data warehouse. We discuss some of these factors here:

Hardware – The clock speed of the hardware, as well as the number of processors, will obviously affect how quickly the test will perform. Additional factors might include the speed and amount of standard and cache memory available to the test process.

Software – Advanced techniques such as sort–merges are legitimate competitive advantages that benchmarks are supposed to be measuring, but shortcomings might be obscured by deft manipulation of one of the other factors. One software factor that is important to consider is s*electivity.* This is describes the percentage of "hits" the query successfully makes. Selecting on gender, for example, would in most cases result in about half of the rows being selected. These rows must then be stored into memory. It may provide a significant advantage to perform the query in a particular order. For example: if the query were to need to find all of the males living in Montana with a last name beginning with 'X,' the query should actually be run in the reverse order. Depending makeup of the data, 'Gender = Male' would return approximately 50 percent of the records, 'State = Montana' might return about 2 percent of the rows, and 'Last Name = X-' might return less than 1 percent of the rows. If the table has 10,000 rows and the query is run in the original order, the search for Gender would mean that 5,000 records would have to be read into temporary memory. Then the State portion of the query would be run against the 5,000 and return 100 records. Lastly, the Last Name portion would run and might return just one name. If the query were run in the reverse order, the amount of memory (and possibly time) required is greatly reduced - 100, 2, and 1 row(s) respectively. *Query optimizers* developed by most vendors is another important software factors that should be considered. Query optimizers have been developed to reconstruct queries to run the most efficiently. A query optimizer may determine that it may be faster to run sequentially rather than referring to an index or running against the index alone. So long as these optimizers are general purpose and not built specifically for the benchmark, these should not be considered unfair. However, that is not always the case. Many compilers include benchmark-specific optimizations that never get used in real-world applications; their only purpose is to increase performance on one specific benchmark.

Database design - This is what is referred to as the dimensional vs. relational controversy. How tables are organized (architecture, indexes, etc.) can be manipulated so that they are configured so that the test avoids problem areas or takes the advantage of strengths or any special features. For example, tables might be pre-joined before the test or have a large number of indexes.

Storage Medium – Giving each process its own dedicated memory and data storage will keep contention to a minimum. Also, access speeds to DASD and network resources might be improved by the use of on-board cache or co-processors.

How to Benchmark a Data Warehouse

A data warehouse can be benchmarked using two distinct classes of performance measures: objective measures and subjective measures. Objective measures are those concerned with the attributes of a data warehouse that can be numerically measured. These data warehouse benchmarking measures typically measure speed of loading files, and accessing information within those files. This is designed to show how well they will work in a given environment. For example, how fast does it process input and output? They help in understanding where processing bottlenecks and inefficiencies occur. Because DSS tools will usually generate highly complex queries, subtle changes in application design, database engine behavior, operating system tuning, and hardware platform design can yield significant benefits in performance. Furthermore, the performance criteria of these components must be consistent with the business environment. Performance settings and criteria optimal for one business community may not be the same of another. It is the responsibility of the systems development team and business user community to work together and analyze the performance capabilities and set the appropriate criteria for themselves.

Subjective measures concern the attributes that cannot be numerically measured, but whose measurement the business user community must define. Subjective performance measures are focused on the effectiveness of the Data Warehouse as a business tool and its acceptance within the business community. The system may be fast, efficient and hold the most relevant data; but if it can not provide effective support for decision making, it may not be considered a success. Moreover, each data warehouse project is unique and can only be measured within its organizational context. Subjective success is achieved if the data warehouse:

- Is driven by the business user community that has clearly identified requirements. Since the primary objective of the Data Warehouse is to facilitate in the decision processes of the users, the users must be responsible for driving the end result. Performance expectations should be clearly documented and outlined on an on-going basis throughout the entire development process, beginning with clear business use cases. As part of the systems life cycle, expectations will be added, changed, and dropped. The successful Data Warehouse management group will adopt a methodology that clearly documents and communicates these changes unilaterally between the systems development group and the business user community.

- Adds value to the decision making process, and can be seen to provide value with better and proven results. It attributes to better tangible decision making. For example, are profit margins of a product increasing due to the lower costs associated with better target marketing?

- Can be understood by the business community. The data in the warehouse and the applications used to extract the data must be clearly understood to ensure that they are utilized to the fullest extent. Furthermore, the data must mean the same to all users. For example an algorithm that provides a statistic must be documented in a way that every user can understand.

- Is used by the business user community. If the Data Warehouse does not deliver quality information with integrity that adds value to the business in a way that the business user community is comfortable with, then it will not be used. Problems may include a difficult to use interface, or being unable to customize reports to get the necessary information. Usage statistics, feedback questionnaires, and user interviews are effective methods of gathering information on subjective performance measures and identifying problems to be resolved.

- Provides a better understanding of the forces acting upon the business and how they are related. A successful Data Warehouse provides increased understanding and knowledge due to its ability to view the business enterprise holistically. For example, analysis of enterprise wide purchasing and inventory patterns can illuminate credit risks and cost savings not otherwise detectable.

Lessons to Remember

Benchmarking of a data warehouse project is fundamentally different from those of other data based systems. A data warehouse has a great impact on the effectiveness and the productivity of those who use it. Every data base project is unique and never ending. It is a dynamic system that requires vigilance from the part of those charged with maintaining it. Here are a variety of actions one can take to try to make the best use of benchmarking when designing a Data Warehouse.

1. Make sure you understand your business needs and the rational for the data warehouse project and where the data warehouse fits within the organization.

2. Secure executive and user support for the data warehouse. Assemble a cross-functional steering committee from the user population to develop a set of corporate policies to measure, manage and monitor the data warehouse continuously (Kimball, 1996).

3. Don’t rely of vendor’s benchmarking results. Research hardware and software solutions and tools carefully. Read the fine print. Make sure that the benchmark has been done on standard hardware and released software. Any special tuning or parameter settings must have been disclosed. It is also important that the customer review all of the disclosure information associated with the published results. This is the “fine print” that may reveal options that the customer cannot feasibly recreate – effectively invalidating the results of the test.

4. Chose a benchmark that matches your Data Warehouse.

References

Berson, A. & Smith, S. (1997). Data Warehouse, Data Mining, & OLAP. New York: McGraw-Hill.

Inmon, W. H. (1996). Building the Data Warehouse. New York. Wiley & Sons Inc.

Kimball, R. (1996). The Data Warehouse Toolkit. New York: Wiley & Sons Inc.