BENCHMARKING A PETABYTE DATA WAREHOUSE

Francois Raab [1] and Paul Krneta [2]
[1]InfoSizing, Inc., CO,USA [2]BMMsoft, CA, USA

ABSTRACT: With the explosion in electronic data and the desire of organizations to perform analytics on this ever increasing source of raw information, big data warehouses are growing at a phenomenal pace. The need to consolidate and correlate structured and unstructured data at increasingly large volumes is becoming commonplace. As data volumes grow, organizations are looking for new ways to store more data and to perform more complex analysis in less time. In this paper we present the design and execution of the first performance benchmark targeting a petabyte data warehouse. We also explore the lessons learned from this project and how they could be applied to the design of a general purpose big data benchmark.

1. INTRODUCTION

The purpose of the benchmark was to evaluate and independently verify the capabilities of a real-world, commercially available big data platform. While the benchmark was sponsored by the product vendors as marketing collateral, it provided a number of insights toward the design of a generic benchmark for big data solutions. This paper focuses on the mechanics of the benchmarking process and looks toward the design of a standard big data benchmarks.

2. MEASURED PLATFORM

The benchmark was executed against a hardware and software solution combining products from multiple vendors. The targeted server was a Sun Enterprise M9000 running the Solaris™ 10 OS and connected to three Sun StorageTek™ 6540 storage arrays. The database manager was the SAP Sybase® IQ analytic server driven by the BMMsoft EDMT ® Server.

3. BENCHMARK OUTLINE

The database schema was designed to accept traditional data and unstructured data. The benchmark took into consideration two key aspects of big data solutions: loading and querying.

3.1 Load Tests

With rapidly increasing levels of data production, the ability of the data warehouse to accept new data feeds should be an integral part of the measure of goodness of a big data solution. Two tests, with several measurement points were defined.

The first data load test focused on recording the load speed for tables hosting traditional data types (values, strings, dates, etc.) In total, one petabyte of raw data was loaded. This data represented individual records from 6 trillion stock quotes, as would be seen from the ticker of a major market exchange. The load process took place in multiple stages with measurement points at each stage of the load.

The second data load test focused on recording the load speed for tables hosting unstructured data types (email, documents, multimedia, etc.) The data source for these tables was simulating the real-time recording of electronic communications between various market makers. This data represented the full content of blogs, text and email messages, including their attachments. In total, over 72 terabyte of unstructured data was loaded. This data represented 185 million full-text messages and attached files. This second load process was also decomposed in multiple stages to allow for individual measurement points along the way.

3.2 Query Tests

The data analysis capabilities were tested by executing real-life business queries during the load process, between each phase of the load process and against the fully loaded data warehouse. Queries were run from 1 to 50 concurrently executing streams. Three types of query tests were run.

The first type of tests measured query times against a static data warehouse. These tests were intended to obtain a baseline from which to evaluate other measurements. Decomposing the load into phases gave an opportunity to obtain query execution times at various “quiet” stages between load phases. As a result, the benchmark produced performance data along a range of data warehouse sizes and provided insight into the effect of data set size on query execution.

The second type of tests measured query times against a data warehouse at various stages in the load process. The main focus of these tests was two folds: first to evaluate how an on-going data load impacts query performance; and second to evaluate changes in query performance as the volume of the data warehouse increases.

The third type of tests measured the visibility delay or "ready-time" of newly loaded data. The purpose of these tests was to execute queries targeting a logical data set that included records or documents being actively loaded into the data warehouse. By repeating the execution of these queries and examining their results, it was possible to measure the delay between the introduction of a new piece of data and its presence in a query result. In other words, the test measured how long it took for new data to become visible to the on-going flow of queries.

4. BENCHMARK RESULTS

In all benchmark exercises, the data that is collected falls in two categories. The first category consists of what is usually referred to as "engineering data". This information represents the perennial submerged part of the iceberg. While some small portion of high visibility data is publicly reported, most of the information gathered during benchmark testing is kept from disclosure and used by the various engineering teams to improve their products.

The second category of information, the visible part of the iceberg, is used for the public disclosure of the benchmark results. The petabyte benchmark followed this model.

The following high-level results were collected during the benchmark.

- The load rate for tables holding traditional data types was 3 million rows per second, or 260 billion rows per day. This rate was not significantly affected by the size of the data warehouse and was maintained for a period of 3 weeks without any manual re-structuring of the data warehouse.
- The load rate for tables holding unstructured data types was 775 documents per second, or 67 million documents per day. This rate was not significantly affected by the size of the data warehouse and was maintained for a period of 3 days without any manual re-structuring of the data warehouse.
- Query execution times were not significantly affected (below 5%) by increases in data warehouse sizes or by the number of query streams.
- During the loading of unstructured data, the ready-time of newly loaded documents was consistently under 5 seconds. This time included in-flight capture, document analysis, table insertion and query response. Increasing the size of the data warehouse or the number of concurrent query streams had less than a 5% impact on the ready-time.

5. CONCLUSIONS

When defining a benchmark, care must be given to the definition of the workload, the execution rules, the environmental constraints, the data collection points and how the results are aggregated into metrics. The petabyte benchmark was no exception. Since no prior model could be followed or extrapolated, all aspects of the benchmark had to be carefully planned and new metrics had to be defined. But, the existence of a standard big data benchmark would have eliminated this planning phase and all but guaranteed the value of the outcome, with the substantial benefit of inherent comparability with other results.

ABOUT THE AUTHORS

1. Francois Raab is President of InfoSizing. As the original author of TPC-C, Raab brings 25 years of experience in defining industry standard performance benchmarks and validating benchmark results. Prior to creating InfoSizing, he developed and taught classes on database internals and tuning.

2. Paul Krneta is the CTO of BMMsoft. He served as CTO for Sybase IQ where he designed IQ Multiplex (shared disk MPP DW), NonStopIQ and designed two world's largest DWs. As Technical Director for Database Technology at Digital (DEC) he optimized Oracle to be the first 64-bit database capable of VLM ("Very Large Memory") – the in-memory database.