Benchmarking Infrastructure for Big Data

Stephen Daniel
NetApp
April 17, 2012

Abstract

This paper discusses the applicability of 20 years of experience benchmarking transactional systems (TPC) and storage system benchmarking (SPC) to the design of systems suitable for use in comparing different analytic systems.

One of the most challenging problems is finding tools to make meaningful comparisons between substantially different architectures working on solving similar problems. For example, architects have tackled problems of poor CPU utilization in large Hadoop clusters by using different network topologies, adding high-performance storage, or using widely varying system architectures.

Examining past benchmark experience highlights strategic choices: should the basis of the benchmark be a defined problem to be solved (TPC) or a defined executable to run (SPC)? How can we have enough realism in the benchmark definition so as to prevent the results from collapsing into a single dimension (TPC-C)? Do benchmarks that mandate publishing pricing encourage more realistic hardware configurations?

This white paper lays out the questions that any successful system benchmark must address and makes a set of proposals for the structure of benchmarks suitable for use in comparing various implementations of Big Data.

Introduction

The increasing availability of low cost solutions for analyzing massive data sets has led to an explosion in "Big Data" and in systems to analyze Big Data. Early work on hardware solutions for large scale analytic systems focused on using low cost, commodity hardware components coupled with inexpensive software that enables very large scale clusters.

As the systems have gone into production and continued to grow interest in cost minimization has shifted from simply focusing on inexpensive hardware towards a systematic approach to minimizing the total cost a large-scale analytic solution.

The shift from $/server to $/total solution has opened the doors to a number of novel solutions. We see various software vendors claiming to offer much better performance per CPU. Storage architectures range from 2 disks per server to 24 JBOD disks per server to various mixes of shared storage. Some vendors are demonstrating excellent price/performance for some workloads using solid state disks (SSDs) for some or all of the storage stack.

Faced with competing solutions with competing claims for cost reduction end users are hard-pressed to evaluate competing claims and vendors struggle to know if they are offering real value.

The rapid pace of development of analytic systems will drive the development of standardized workloads and benchmarks to reduce the confusion and permit competition on the basis of measurements rather than theoretical arguments.

Unfortunately experience has shown that development of meaningful and broadly useful benchmarks is also a complex endeavor. Drawing on the history of the Transaction Processing Performance Council (TPC) and the Storage Performance Council (SPC), this paper attempts to translate some of the lessons learned in previous multi-vendor benchmark consortiums into useful advice for those wishing to create standardized ways to evaluate analytic systems suitable for Big Data.

Benchmarks and Load Generators

There is a useful distinction between a performance benchmark and load generator. Load generators create a well-defined set of work to be performed. They are often tunable (e.g. IOMeter). By itself a load generator is not sufficient to generate reliable comparisons between widely different implementations. A performance benchmark includes a load generator, a set of run-rules for the generator, requirements on reporting the result, limitations on optimizing for the benchmark, and regulatory oversight on use of results.

This difference can be seen by comparing Dhrystone, a popular tool for measuring CPU performance, and TPC-C, a popular system benchmark. Dhrystone is widely derided; vendors are known to lie or position results in misleading ways[i]. The TPC-C benchmark is technical obsolete, but because of strong control by the TPC the issues with the benchmark are confined to its technical relevance to today's market. Vendors are under a tight leash that minimizes the amount of lying and distorting involved in presenting the results.

The TPC achieves this result by requiring that results that are independently audited, have pricing, are reviewed and can be challenged. Any public use of the results is also regulated and subject to challenge[ii]. In contrast, Dhrystone is simple a program that anyone can run.

Specifying a Workload

The TPC concerns itself with benchmarking systems for transaction processing. Their benchmarks are specified as a set of work to be accomplished. Vendors wishing to publish a TPC benchmark must acquire an implementation of the benchmark. A significant portion of the audit and review cycle goes to ensuring the implementation actually conforms to the specification.

The Storage Performance Council (SPC) is a similar body, developing benchmarks for storage systems. The SPC was modeled on the TPC and shares many attributes. One substantial difference is that the SPC creates and regulates workload generators, not just workload specifications. To produce a conforming result a vendor must comply with the specification, and must also use the regulated workload generator.

This difference has profound implications for the way the two bodies operation. The TPC continually reviews implementations and struggles to ensure that implementations are conforming. The decisions are inevitably based on a mixture of technical correctness and vendor self-interest. In contrast in most years the SPC's compliance review committee has reviewed zero complaints.

While the SPC's approach clearly has produced much less controversy, it comes at a cost. The SPC can only benchmark storage systems that conform to a conventional definition of a LUN. The TPC can theoretically benchmark anything that runs the workload. The TPC's flexibility allows them to compare a novel and new database technology to a 30-year old mainline product.

Recommendations for Analytics Benchmarks

1. We need well-structured benchmarks, supported by active organizations which will not only create but maintain the benchmarks as technology evolves.
2. We need requirements for full disclosure, including pricing.
3. We need a publication process that includes audit, peer review, and fair use of results. The TPC and SPC both provide examples of strong systems for building these frameworks.
4. We should specify workloads, not workload generators. This is harder, will take longer, and will generate more conflict and controversy; however, it will enable comparison across a much more diverse set of solutions.
5. If we want reasonable time-to-market we should expect to pay for help managing and editing benchmark specifications.
6. We should expect to need multiple benchmarks covering disjoint application areas.

Conclusions

Having viable benchmarks for competing analytics solutions is both possible and desirable. In creating such benchmarks we should learn from the success of the TPC and SPC – benchmarks require a continuing support organization. They cannot survive as a body of code simply published for all to use.

[i] http://ww2.cs.mu.oz.au/313/online/Handout.pdf, slide set 8, number 11

[ii] http://www.tpc.org/information/about/documentation/TPC_Policies_v5.23.htm, sections 6 and 8