Development of a Performance Database Web Server Tailored to User Services Needs

Richard Marciano, David Ence, and *Bahram Nassersharif,* National Supercomputer Center for Energy and the Environment, University of Nevada, Las Vegas, Las Vegas, NV

Abstract: *Our user services experience at the National Supercomputing Center for Energy and the Environment, NSCEE, has demonstrated the need to keep users informed on machine performance issues. This requires special user support awareness and training. To better respond to this need, we are developing a performance database system allowing a comparative analysis of benchmarks including ease of use, portability, scalability, limitations, meaningfulness of results, among others. We call this tool BONUS: Benchmark-Oriented Navigator for User Services. BONUS is based on a general and scalable database model of benchmark characteristics. It is designed to meet user services needs, allow the browsing/plotting of machine/benchmark data, and allow easy referencing of associated papers and technical reports. This paper is meant to serve as a blueprint for the construction of the BONUS system.*

1 Introduction

Over the last year, the User Support staff at NSCEE, has had to respond to an increasing number of requests dealing with comparative performance measurement. Typical questions range from the naive "How fast will this run on the supercomputer?", and "Why should I use a supercomputer when it runs almost as fast as my workstation?", to more sophisticated inquiries where users have gone through the trouble of benchmarking some kernels or application codes across selected platforms and wonder about the results obtained.

User Support performance-reporting pitfalls include quoting numbers which the user may take out of context, and overselling or underselling the machine, opening the way for disillusionment and frustration over the inability to achieve comparable performance rates.

1.1 Training Solutions

To better service performance-based requests, NSCEE has initiated the following training activities:

1. Writing a series of articles in *Terabit*, NSCEE's newsletter, on benchmarking techniques and specific benchmark tests (HINT, Parallel NAS, ParkBench, Linpack, etc.)
2. Dowloading and making specific benchmarks available to users (Linpack, HINT, etc.)
3. Reporting on actual results obtained, not just on vendor-supplied figures
4. Staff training on the significance of benchmark results, the complexity of reporting results, the need to put them in context
5. User support training through performance monitoring utilities, classes, code optimization workshops, vectorization/parallelization seminars, and the offering of computational science graduate courses.

While these activities appear to be an effective way of disseminating performance knowledge, their effects are more long-term, and the audience they reach more limited. Moreover, the user technical support staff stills bears the responsibility of responding directly and synthetically to performance queries.

1.2 User Support Dilemma

We illustrate the dilemma of providing user support on benchmarking/performance issues on the following representative performance query. Our staff was approached with a request for assistance concerning two separate rankings of the top 19 machines on campus. Two programs were used to establish these lists: the first, a 32-bit integer benchmark calculating Pi, the second, an in-house 64-bit floating point general 2-D finite difference program. Results listed the HP 9000/735 and SGI Onyx as outperforming the Cray Y-MP2/216. Based on these results, several researchers approached us with grave concerns and expressed reservations about whether they should even bother to use supercomputers for their applications!

Note: In this paper, we define a *benchmark* as a program used to evaluate the performance of a wide range of computer systems. "What distinguishes a benchmark from an ordinary program is a general concensus of opinion within the industry that the benchmark exercises a computer well" (Conte [7]).

One of the dilemmas faced by user support staff is the responsibility to address such performance concerns without having to completely revalidate the outside benchmarking study. This would be very time-consuming and require expert knowledge of the benchmark application code area (finite difference code using a pure implicit method in this case). Bailey [3], sums this up quite well in his "Twelve Ways to Fool the Masses When Giving Performance Results..." article. It turns out that two of these misleading "ways" were used in compiling the top 19 rankings: (1) *quote 32-bit performance not 64-bit*, (2) *compare your results against scalar, unoptimized code on the Crays*.

In summary, benchmarking is a non-trivial activity, on occasion even described as an art form! This is reflected in the titles of many studies: "Needed: A Measure for Performance" (Johnson [17]), "How not to Lie with Statistics..." (Fleming [9]), "Benchmarking: Can There Be an Industry Standard" (Hill [13]), "The Art of SPECmanship" (Bradley [6]), "The Mainframe Performance Debate" (Moad [25]).

To make matters more painful, if one takes a closer look at the various benchmark programs frequently cited, a partial alphabetical list would run as follows (take a deep breath!): *007 (ODBMS), ADPAC, AIM, ANSYS, APB330, Abaqus, B501B, BC, BKKConf, BPolm, Baro, Bonnie, Business Benchmark, Byte, CDNS, CHARMm, CPU2, DHele10, Dhrystone, Digital Review, Dodoc, Dynad3D, EDMC, EDN, ENSAERO, EWave, Esn, EuroBen, FFT, Fhourstones, Fidap, Flops, Fluent, GAMESS, GPC, GammaF, Gamteb, Gaussian 92, Ghraphstones, Hanoi, Hartstone, Heapsort, Hydro, IO, IOBENCH, IOZONE, Intmc, KT, Kawab, Khornerstone, Kimpl, LFK (Livermore Loops), LSS, Linflox, Linpack, Los Alamos Benchmarks, MHD2D, MIPS, MSB4, MUSBUS, Matrix Multiply (MM), Mclrec, Mclvsi, NAS Kernels, NAS Parallel Benchmarks, Nastran, N-body, Netperf, Nettest, Neut, Nhfsstone, NIKE2D, OGCM, OVERFLOW, PC Bench/WinBench/NetBench/iCOMP, PLB picture-level benchmark, PVSOLVE, Perfect Club, Photon, Prgmx, Pueblo, RSGaas, Rhealstone, RhosettaStone, SCF, SCStart, SLALOM, SPARK3D, SPEC benchmarks, SPICE, SSBA, SYSmark, Sieve of Eratosthenes, Sim, Stanford, Star-CD, TFFTDP, TPC, Ttcp, TurboKiva, Twodant, Ulcont, Unichem, VGam, Vecops, Vecskip, Vortex, WPI Benchmark Suite, Wave, Whetstone, X3d, Xstone, Zelig!*

The adopting of a reduced set of metrics is desperately needed! This is precisely the goal of the JNNIE project (JNNIE [16]), the intent of which is to define a single set of evaluation parameters to better evaluate the effectiveness of scalable parallel computing systems (SPC's) under credible scientific workloads. Other related studies include the GENESIS (Papiani [26]) and ParkBench (Hockney [15]) benchmarks. These recent attempts at standardization, however, primarily focus on the new and emerging parallel machines and do not deal with cross-technology comparisons (workstation vs. supercomputer, etc.), nor do they offer a practical solution for the user support dilemma.

2 The *BONUS* Approach

"The process of gathering, archiving, and distributing computer benchmark data is a cumbersome task usually performed by computer users and vendors with little coordination" (Hockney [15]). After surveying the literature and measuring needs from a User Services perspective, we established the need for a comprehensive performance database repository, and the development of a "user services"-oriented browsing/querying tool that allows remote Internet access and control. We call this tool ***BONUS***: Benchmark-Oriented Navigator for User Services.

2.1 General Requirements

Such a tool should:

- Be based on a general and scalable database model of benchmark characteristics
- Be designed to meet user services needs
- Allow the browsing/plotting of machine/benchmark data
- Allow easy referencing of papers, technical reports, and source code

2.2 Benchmark Characterization

The characterization of common properties between benchmarks, and features leading to a possible taxonomy is an essential first step. Several studies provide partial benchmark characterizations (Berry [5], Conte [7], JNNIE [16], Price [29]). Benchmark performance can vary according to the memory hierarchy, cache size, etc. Follow, important benchmark attributes which must be recorded in order to make results reproducible and meaningful:

General Attributes

Benchmark Suite & Revision:

Benchmark:

Type: kernel, local, partial, recursive, synthetic, application

Nature: computational, data base, network, graphics, I/O

Usage: workstations, PCs, supercomputers, parallel computers

Intent: serial, parallel language, shared memory, message-passing

Reporting Metrics: MIPS, MFIPS, Mflops/s, tps, QUIPS, packets/s, graphics, SPECMark ratio, time (elapsed, io, cpu), PLB, GPCmarks

Code Details

Lines of Code: (thousands)

Discipline: weather, chem, bio, env, phys, CFD, etc.

Algorithms: dense linear algebra, sparse linear algebra, finite differences, eigensystems, stochastic models, sorting/searching, optimization, numerical integration

Code Origin: research group/public domain

Dominant Computation: floating point, integer, logical

Language Implementation: assembly, Fortran, C, Pascal, pencil & paper

Test Configuration

- Hardware

Manufacturer:

Model number:

CPU:

CPU speed (MHz):

FPU:

Primary cache:

Secondary cache:

Other cache:

Memory:

Disk subsystem:

Network:

Other:

- Software

O/S & version:

Compiler revision:

File system type:

Libraries used & version:

Compiler switches:

- System

Tuning parameters:

Background load:

System state:

- Comments

Benchmark Configuration

Date benchmarked (DD/MM/YY):

Problem size (if applicable):

Number of nodes (if applicable):

These benchmark attributes form the backbone of the benchmark repository system we are proposing. ***BONUS*** should allow the perusal and search of these attributes and allow queries of the nature: "show me all the benchmarks that are intended to target message-passing machines".

2.3 Meeting User Services Needs

Additional attributes need to be defined that are of direct interest to user services staff. These attributes needn't merely be quantifiable and should include **subjective measures** as well. We propose the following attributes, each rated on a scale of 1 to 10:

- ease of use
- portability
- scalability

Each site could define its own local "subjective" parameters.

The system should also allow the support staff person using the tool to update his/her own statistics. These could include the number of user-related requests for a particular benchmark, a simple **counter attribute** could be maintained.

Finally, the system should be **user-friendly**. We propose a World Wide Web solution. Most online information resources are being made available through Web servers, allowing the convenience of formatted text, embedded with hyperlinks and graphical images.

2.4 Browsing and Plotting of Data

The need for browsing and data plotting features has been well illustrated in earlier performance reporting systems such as GBIS (Papiani [26]) and PDS (Hockney [15]). The ability to list available vendors and benchmarks is needed as well as the need for visually meaningful plots of matching performance data. GBIS allows output plots to be generated on the fly in a variety of formats: gif, xbm, postscript, or tabular. Whenever possible, the user should be able to specify axes types (log or linear) and plotting ranges. GBIS manages to give a consistent user-interface to its plotting component, owing to the focus of the underlying benchmarks: message-passing only.

BONUS proposes to apply a mix of tabular and graphical output formats, reflecting the mix of architecture classes and benchmark types it purports to capture.

2.5 Cross-Referencing

Finally, the need for associated references on benchmarking is apparent. PDS approaches this issue by "providing abstracts and complete papers related to benchmarks and thereby providing a needed educational resource without risking improper interpretation of retrieved benchmark data (Hockney [15]). We propose to make this an integral browsing feature. References to source code location are also needed.

3 *BONUS* Status

The Benchmark-Oriented Navigator for User Services we are proposing is in its first phase of development. Phase I consists of collecting, porting, and installing benchmarks directly to our local systems. Benchmark attributes are recorded and the database fields we proposed in Section 2 are being validated.

In parallel, we are exploring the optimal Web interface and ways to browse/plot data more effectively. It is expected that a working prototype will be "handed" over to user services within the next two months for evaluation and early use.

Rather than serving as a monolithic and centralized benchmark repository, NSCEE wishes to validate this new approach, allowing any supercomputing site to set up its own performance database server, based on its own specific needs and users. The tools and methodology being developed are expected to port to a variety of systems and environments.

4 Summary

Several recent systems attempt to gather, archive, and distribute computer benchmark data. The information one can browse, however, is not based on a "universal" performance data database model. This paper attempts to define and capture the attributes that characterize a broad range of benchmarks, not just a specialized subset. The browsing, searching, and updating of all of these attributes is made possible by a tool called ***BONUS***. ***BONUS*** is a web-based tool specifically targeted to the needs of user services staff. It allows the browsing and plotting of specific data as well as the cross-referencing of abstracts and papers, thereby serving as an important educational resource.

References

1. Bailey et al., *"The NAS Parallel Benchmarks (94)"*, RNR Technical Report RNR-94-007, March 1994.
2. Bailey, D., Barszcz, E., Dagum, L., Simon, H. D., "NAS Parallel Benchmark Results", *IEEE Parallel & Distributed Technology*, Feb. 1993, pp. 43-51.
3. Bailey, D., "Twelve Ways to Fool the Masses When Giving Performance Results on Parallel Computers", *Supercomputing Review*, Aug. 1991.
4. Bell, G., "Scalable, Parallel Computers: Alternatives, Issues, and Challenges", *International Journal of Parallel Computing*, Vol. 22, No. 1, Feb. 1994
5. Berry, M., Cybenko, G., Larson, J., "Scientific Benchmark Characterizations", *Parallel Computing*, Vol. 17, 1991, pp. 1173-1194.
6. Bradley, B., "The Art of Specmanship", *Workstation News*, Jan. 1991, pp. 28-29.
7. Conte, T. M., Hwu, W. W., "Benchmark Characterization", *IEEE Computer*, January 1991, pp.48-56.
8. Evans & Sutherland, *"High-End Graphics Workstations: A Competitive Report"*, Technical Report by Evans & Sutherland, May 21, 1993.
9. Fleming, P. J., Wallace, J. J., "How not to Lie with Statistics: The Correct Way to Summarize Benchmark Results", *Communications of the ACM*, March 1986, Vol. 29, No. 3, pp. 218-221.
10. Frenkel, K. A., "Evaluating Two Massively Parallel Machines", *Communications of the ACM*, Aug. 1986, Vol. 29, No. 8, pp. 752-758.
11. Gee, J. D., Hill, M. D., Pnevmatikatos, D. N., Smith, A. J., "Cache Performance of the SPEC92 Benchmark Suite", *IEEE Micro*, Aug. 1993, pp. 17-27.
12. Gustafson, J., Snell, Q., *"HINT: A New Way To Measure Computer Performance"*, Technical Report, Ames Scalable Laboratory, http://www.scl.ameslab.gov/scl/HINT
13. Hill, R. H., "Benchmarking: Can There Be an Industry Standard?", *Supercomputing Review*, January 1990, pp. 50-53.
14. Hinnant, D., "Performance Measures: Benchmarks Are Only One Facet of the Broad and Complex Field of Computer Performance Evaluation", *UNIX Review*, Vol. 8, No. 12, pp. 34-40.
15. Hockney, R., Berry, M., *"Public International Benchmarks for Parallel Computers"*, PARKBENCH Committee: Report-1, Feb. 7, 1994
16. JNNIE Project, Joint NSF-NASA Initiative on Evaluation, pronouced "genie", see http://www.tc.cornell.edu/JNNIE/JNNIEHome.html
17. Johnson, R. R., "Needed: A Measure for Measure", *Datamation*, Dec. 15, 1970, pp. 22-30.
18. Jordan, K. E., *"Performance Comparison of Large-Scale Scientific Computers: Scalar Mainframes, Mainframes with Integrated Vector Facilities, and Supercomputers"*, Computer Magazine, Los Alamitos, CA, March 1987.
19. Kang, Y. M., "Computer Hardware Performance: Production and Cost Function Analyses", *Communications of the ACM*, May 1989, Vol. 32, No. 5, pp. 586-593.
20. Karp, A. H., Flatt, H. P., "Measuring Parallel Processor Performance", *Communications of the ACM*, May 1990, Vol. 33, No. 5, pp. 539-543.
21. LaRose, B., *"The Development and Implementation of a Performance Database Server"*, University of Tennessee Technical Report CS-93-195, August 1993.
22. "MIPS Ratings' Rise & Fall", *Workstation News*, Jan. 1991, pp. 25-26.
23. Machrone, B., "Benchmark Test Progress", *PC Magazine*, March 16, 1993, pp. 87-88.
24. Magney, P., "Read With Caution: Benchmarks Are Valuable Only if They're Taken in Context", *Workstation News*, Jan. 1991, pp. 27-28.
25. Moad, J., "The Mainframe Performance Debate", *Datamation*, Jan. 15, 1992, pp. 47-48.
26. Papiani, M., *"Genesis Benchmark Information Service (GBIS) Graphical Interface"*, http://hpcc.soton.ac.uk/RandD/gbis/papiani-new-gbis-top.html
27. Parker Smith, N., *"GLOPS, BLOPS and PLOPS: An End to Performance Confusion"*, HPCwire electronic magazine, Feb. 17, 1995.
28. Presont, K., "The Abingdon Cross Benchmark Survey", *IEEE Computer*, July 1989, pp. 9-18.
29. Price, W. J., "A Benchmark Tutorial", *IEEE Micro*, Oct. 1989, pp. 28-43.
30. "Selected SPEC Results", *Supercomputing Review*, Sep. 1990, pp. 47-57.
31. Simmons, M. L., Wasserman, H. J., Lubeck, O. M., Eoyang, C., R. Mendez, Harada, H., Ishiguro, M., "A Performance Comparison of Four Supercomputers", *Communications of the ACM*, Aug. 1992, Vol. 35, No. 8, pp. 116-124.
32. Smith, J. E., "Characterizating Computer Performance With a Single Number", *Communications of the ACM*, Oct. 1988, Vol. 31, No. 10, pp. 1202-1206.
33. Van-Catledge, F. A., "Toward a General Model for Evaluating the Relative Performance of Computer Systems", *The International Journal of Supercomputer Applications*, Vol. 3, No. 2, Summer 1989, pp. 100-108.
34. Weicker, R. P., "An Overview of Common Benchmarks", *IEEE Computer*, Dec. 1990, pp. 65-75.
35. Worlton, J., "Understanding Supercomputer Benchmarks", *Datamation*, Sep. 1, 1984, pp. 121-130.
36. Zorpette, G., "Finally: a Benchmark for Graphics", *IEEE Spectrum*, July 1991, pp. 18.