



aasish-kumar.sharma@gwdg.de

Aasish Kumar Sharma

Benchmark

High Performance Computing System Administration

Theory: Benchmark

What?

Alert: Standard Vs Benchmark

Standard

- It is a published specification ratified by some organization.
- e.g., ISO 9001 - is an international standard for quality management.

Benchmark

- It is a test to evaluate your system's performance,
- It is often established by general organizational acceptance.
- e.g., IO 500 Benchmark - is a comparison against standards.



Images: <https://www.iso.org/modules/isoorg-template/img/iso/iso-logo-print.gif>,
<https://www.vi4io.org/io500/start>, <https://www.top500.org/news/chinas-tianhe-2-supercomputer-retains-top-spot-on-43rd-edition-of-the-top500-list/>

Types of Benchmarks

Generally, there are six types of Benchmarks

- 1 **Internal**: Comparing processes within the organization,
- 2 **External**: Comparing to other organizations,
- 3 **Competitive**: Specifically comparing to direct competitors,
- 4 **Performance**: Analyzing metrics to set performance standards (in HPC),
- 5 **Strategic**: Evaluating how successful companies strategies,
- 6 **Practice**: Process mapping and addressing performance gaps.

What are Key Metrics of Performance Benchmarking

- **Timing**
 - ▶ It is a measure of full or partial run-time of an application. Mainly wall clock.
- **Baseline performance - Micro Benchmark**
 - ▶ Measures benchmarking performance improvement against unit of node.
- **Scaling - Macro Benchmark**
 - ▶ Measures how the performance changes with the number of nodes/cores.
- **Performance - Overall Benchmark**
 - ▶ It is a measure of rate of how well an application is running.
- **Parallel efficiency**
 - ▶ The ratio of measured scaling to the perfect scaling.

What are HPC Benchmarking Tools and Techniques

Tools and Techniques/Methods

- There are no hard and fast rules for any tools and techniques,
- It entirely depends on the kind of data you have,
- Always choose the tools and techniques that gives the best insights,
- Tools: like graphs/chart and some softwares (e.g., likwid), and
- Methods: like SWOT analysis, Performance Ratio, Life Cycle are helpful.

In Practice

- Mostly, line graphs & bar charts are used for comparisons,
- Whereas, Pie charts are best for showing distributions,

More on: https://en.wikipedia.org/wiki/List_of_benchmarking_methods_and_software_tools

What are Popular HPC Benchmarks

■ HPLinpack & DGEMM

- ▶ Software library for performing numerical linear algebra on digital computers,
- ▶ The most famous HPC benchmark – used for the “Top500” ranking.

■ STREAM

- ▶ Measures “Sustainable Memory Bandwidth”, for COPY, SCALE, ADD, & TRIAD operations.

■ HPC Challenge

- ▶ Extension of HPLinpack benchmark to a suite of benchmarks. e.g., DGEMM.

■ Graph 500

- ▶ A new benchmark for supercomputers based on large-scale graph analysis.
- ▶ An attempt to provide an alternative to HPLinpack for graph problems.

■ NAS Parallel

- ▶ Targets performance evaluation of highly parallel supercomputers.

<https://wgropp.cs.illinois.edu/courses/cs598-s15/lectures/lecture03.pdf>

Benchmark Example

IO-500 10-node Challenge - Bandwidth

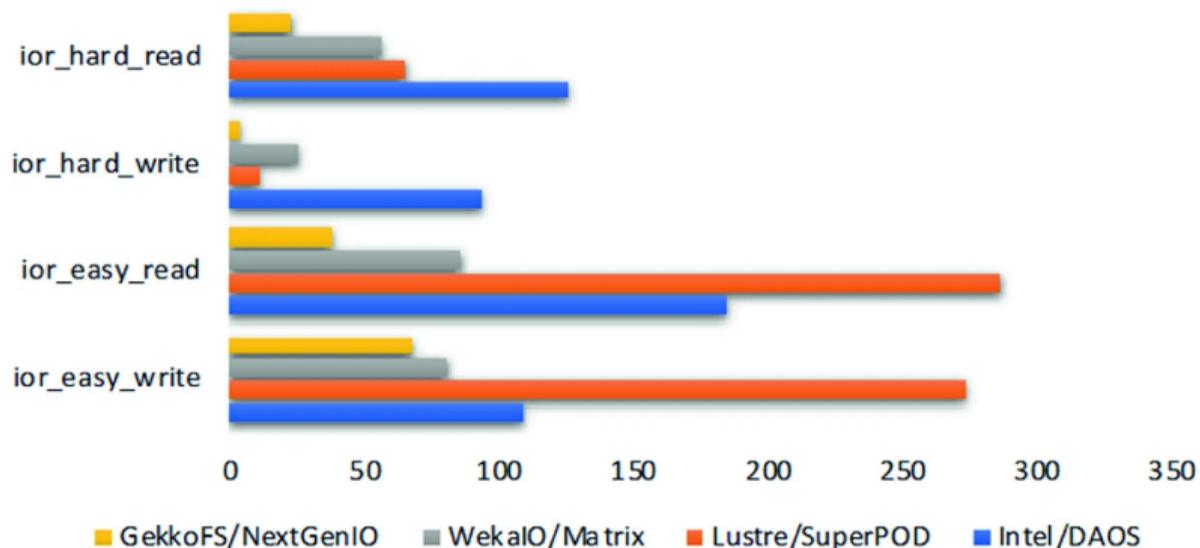


Figure: IO500-IOR bandwidth of the top four storage systems on Nov 2019 edition.

Image: https://link.springer.com/chapter/10.1007/978-3-030-48842-0_3

Theory: Benchmark

Why?

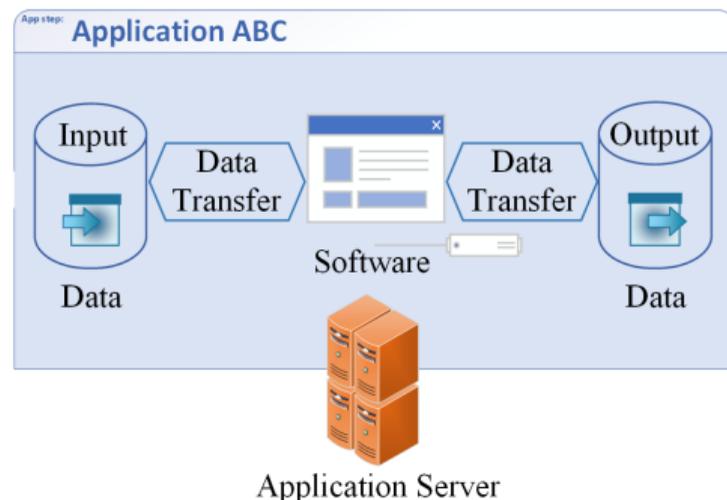
Why We Need Benchmarking in HPC?

To understand:

- How the current systems (node) are functioning. i.e., system calibration.
- How potentially in future the work could scale so that the correct amount of resources are allocated to applications.

Note:

- Application means
 - ▶ Software running on the system, &
 - ▶ Input/Output data required by it.
 - ▶ Need benchmarking whole process.



The Challenges for HPC Benchmarks?

Challenges for HPC Evaluation With Realistic Applications

- 1 Simple benchmarks are overly easy to run,
- 2 Realistic benchmarks cannot be abstracted from real applications,
- 3 Today's realistic applications may not be tomorrows applications,
- 4 Benchmarking is not eligible for research funding,
- 5 Maintaining benchmarking efforts is costly,
- 6 Proprietary full-application benchmarks cannot serve as yardsticks.

Theory: Benchmark

How?

How to Fool The Masses

12 Ways to Fool the Masses When Giving Performance Results on HPC

- 1 Quote the internal performance results projected to a full system.
- 2 Quote only 32-bit completely excluding 64-bit performance results.
- 3 Compare the results against scalar, unoptimized code on Crays (HPC).
- 4 Quietly employ assembly code and other low-level language constructs.
- 5 Presenting performance figures of internal test for the entire application.
- 6 Scaling up the problem size with processors, without mentioning this fact.
- 7 Comparing old code on an obsolete system for direct run time requirement.

<https://www.davidhbailey.com/dhbpapers/twelve-ways.pdf>

How to Fool The Masses (Contd..)

12 Ways to Fool the Masses When Giving Performance Results on HPC

- 8 Providing the operation count based on the parallel implementation, neglecting the best sequential one.
- 9 Quote performance in terms of processor utilization, parallel speedups or MFLOPS per dollar.
- 10 Mutilate the algorithm used in the parallel implementation to match the architecture.
- 11 Measure the conventional run times in a busy environment and parallel on a dedicated system.
- 12 If all else fails, show pretty pictures and animated videos, and don't talk about it.

How to Tell The Masses The Performance Benchmarks

12 Ways to Tell The Masses When Reporting Performance Results

- 1 For parallel speedup report, do not miss, a single parallel process, serial execution, or the absolute execution,
- 2 Specify the reasons, for and against, whatever you are reporting,
- 3 Use the arithmetic mean only for summarizing costs and harmonic mean only for summarizing rates,
- 4 Avoid summarizing ratios; use the geometric mean for summarizing ratios when the ratio of base costs or rates are not available,
- 5 Report if the measurement values are deterministic else report confidence intervals of the measurement,
- 6 Do not assume normality of collected data (e.g., based on the number of samples) without checking diagnostic,

Hoefler and Belli, "Scientific Benchmarking of Parallel Computing Systems"

How to Tell The Masses The Performance Benchmarks (Contd..)

12 Ways to Tell The Masses When Reporting Performance Results

- 7 Compare non-deterministic data in a statistically sound way, e.g., ANOVA.
- 8 Carefully investigate if measures of central tendency like mean or median are useful to report else use other measures like percentiles.
- 9 Document everything (e.g., software, hardware, techniques) to facilitate reproducibility and provide interpretability.
- 10 For parallel time measurements, report all measurement, (optional) synchronization, and summarization techniques.
- 11 If possible, show upper performance bounds to facilitate interpretability of the measured results.
- 12 Plot all needed information to interpret the experimental results. Only connect valid interpolation measurements by lines if they indicate trends.

Hoefler and Belli, "Scientific Benchmarking of Parallel Computing Systems"

How To Do A Good Benchmarks?

A Few Considerations:

- Think critically,
- Do not neglect any details,
- Use proper measures and charts,
- Adapt to and address the changes properly,
- Apply repetitive and continuous attempts to collect the data,
- Have an alert and attentive (vigilante) approach to the details.

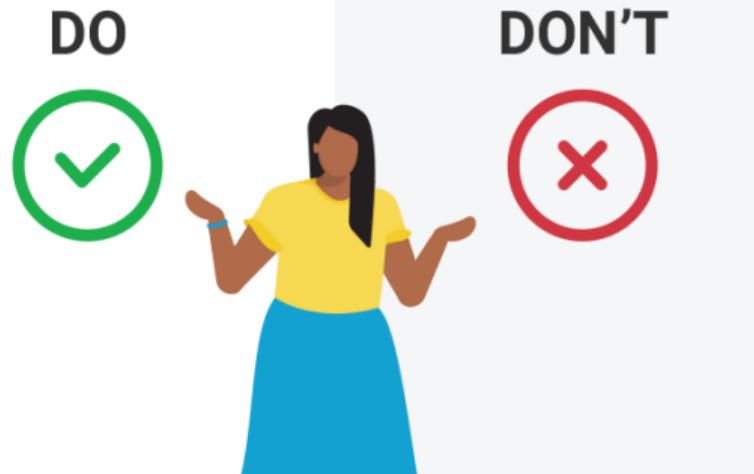


Image: <https://thefruitfultoolbox.com/dos-donts-disc/>

Example: Is Benchmarking?

■ HT - Hyper Threading

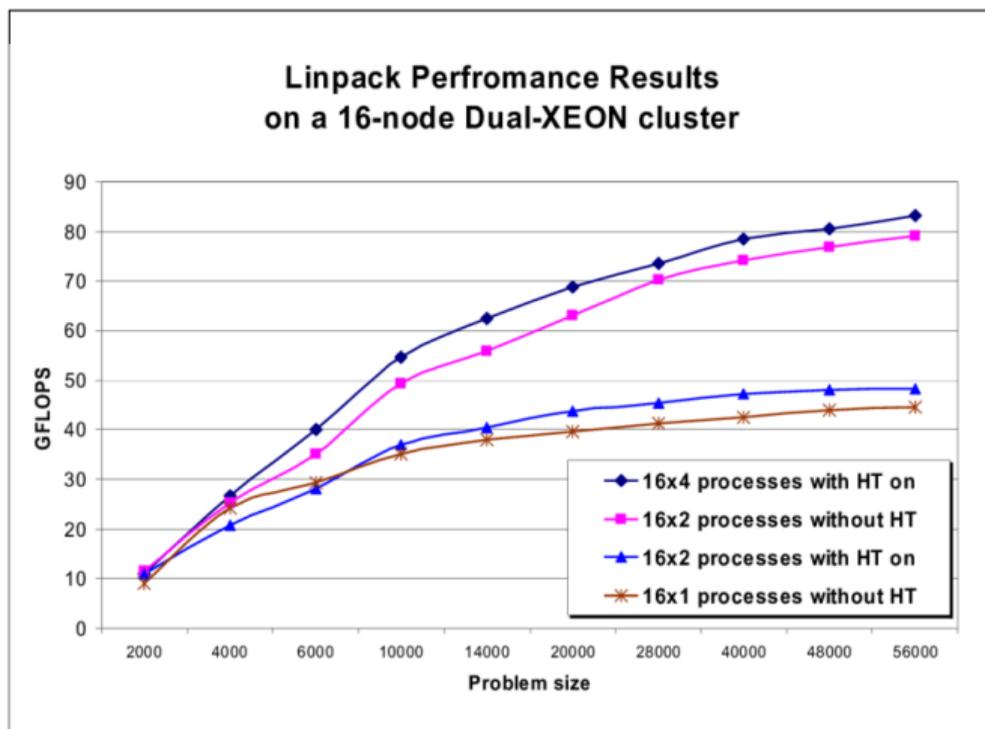


Image: <https://www.researchgate.net/profile/Rizwan-Ali-7/publication/267242498/>

References

- Armstrong, Brian et al. “HPC Benchmarking and Performance Evaluation With Realistic Applications”. In: ().
- Fleming, Philip J. and John J. Wallace. “How Not to Lie with Statistics: The Correct Way to Summarize Benchmark Results”. In: *Commun. ACM* 29.3 (Mar. 1986), pp. 218–221. ISSN: 0001-0782. DOI: 10.1145/5666.5673.
- Hoefler, Torsten and Roberto Belli. “Scientific Benchmarking of Parallel Computing Systems: Twelve Ways to Tell the Masses When Reporting Performance Results”. In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. SC '15*. New York, NY, USA: Association for Computing Machinery, Nov. 2015, pp. 1–12. ISBN: 978-1-4503-3723-6. DOI: 10.1145/2807591.2807644.

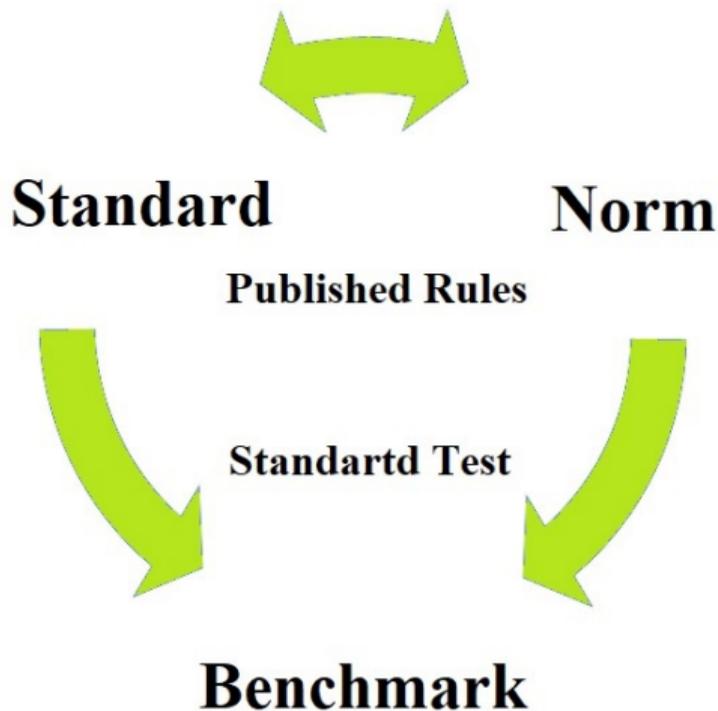
Appendix I

Standards!

- But what is the standard?

Theoretically

- Standards are published rules.
- Prerequisites for a standard are,
 - ▶ should be technically mature, &
 - ▶ should has a benefit for the user.



Appendix II

Standards! (contd..)

- For example,

British Standards (formerly “British Standards Institute”):

- “A standard is a publicly available technical document that is developed with the participation of all interested parties and has their approval. The standard is based on results from science and technology and aims to promote the common good.”

Appendix III

HPC Benchmarking

- Benchmarking is done by assessing the relative performance by,
 - ▶ running a computer program (micro benchmarking), or
 - ▶ a collection of programs (macro benchmarking), or
 - ▶ other operations (overall benchmarking),
- It measures how the system performance with the changes in parameters.
- Normally, it runs a number of standard tests and trials against it.