Performance Analysis of HPC Virtualization Technologies within FutureGrid

Andrew J. Younge, James T. Brown, Robert Henschel, Judy Qiu, Geoffrey C. Fox
Pervasive Technology Institute, Indiana University
2719 E 10th St., Bloomington, IN 47408, U.S.A.
Corresponding Email: {ajyounge,jatbrown,henschel,xqiu,gcf}@indiana.edu

***Abstract*—As Cloud computing emerges as the dominant paradigm in distributed systems, it's important to fully understand the underlying technologies that make clouds possible. One technology, and perhaps the most important, is virtualization. Recently virtualization through the use of hypervisors has become widespread and well understood by many. However, there are a wide spread of different hypervisors, each with their own advantages and disadvantages. This emerging research provides a full analysis of today's best virtualization technologies from feature comparison to in-depth performance analysis, focusing on the applicability to high-performance computing environments using FutureGrid resources.**

***Index Terms*—Cloud Computing, Virtualization, Hypervisor, FutureGrid**

I. Introduction

Cloud computing is one of the most explosively expanding technologies in the computing industry today. A Cloud computing implementation typically enables users to migrate their data and computation to a remote location with minimal impact on system performance [1]. This provides a number of benefits which could not otherwise be realized. These benefits include scalability, enhanced quality of service, a specific, custom and specialize environment for the users, cost effectiveness through economies of scale, and a simplified interface to access the resources.

There are a number of underlying technologies, services, and infrastructure-level configurations that make Cloud computing possible. One of the most important technologies is the use of virtualization [2], [3]. Virtualization is a way to abstract the hardware and system resources from a operating system. This is typically performed within a Cloud environment across a large set of servers using a Hypervisor or Virtual Machine Monitor (VMM) which lies in between the hardware and the Operating System (OS). From here, one or more virtualized OSs can be started concurrently as seen in Figure 1, leading to one of the key advantages of Cloud computing. This, along with the advent of multi-core processing capabilities, allows for a consolidation of resources within any data center. It is the Cloud's job to exploit this capability to its maximum potential while still maintaining a given QoS.

Virtualization is not specific to Cloud computing. IBM originally pioneered the concept in the 1960's with the M44/44X systems. It has only recently been reintroduced for general use on x86 platforms. Today there are a number of Clouds that offer IaaS through the use of virtualization technologies. The Amazon Elastic Compute Cloud (EC2) [4], is probably the most popular of which and is used extensively in the IT industry. Eucalyptus [5] is becoming popular in both the scientific and industry communities. It provides the same interface as EC2 and allows users to build an EC2-like cloud using their own internal resources. Other scientific Cloud specific projects exist such as OpenNebula [6], In-VIGO [7], and Cluster-on-Demand [8], all of which leverage one or more hypervisors to provide computing infrastructure on demand. Using a Cloud deployment overlaid on a Grid computing system has also been explored by the Nimbus project [9] using Globus. The virtualization technique of choice for these open platforms over the past 5 years has typically been the Xen hypervisor [2], however more recently VMWare [10] [1], Oracle VirtualBox [11] and the Kernel-based Virtual Machine (KVM) [12] have become commonplace.

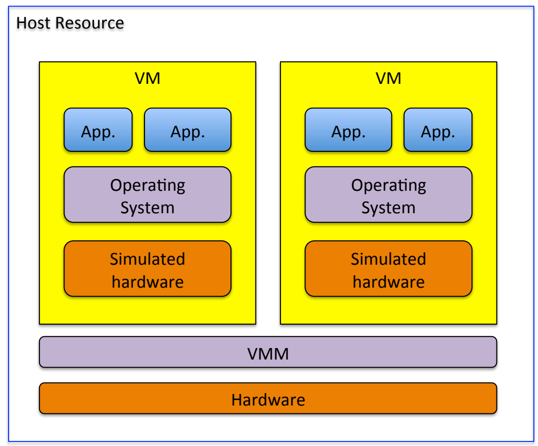

Fig. 1. Virtual Machine Abstraction

As these underlying hypervisor and virtualization implementations have evolved rapidly in recent years along with virtualization support directly on standard x86 hardware, it is necessary to carefully and accurately evaluate the performance implications of each system. As such, we conduct an investigation of several virtualization technologies, namely Xen, KVM, VirtualBox. Each hypervisor is compared alongside one another with base-metal as a control and run through a number of High Performance benchmarking tools.

II. Experimental Design

A. The FutureGrid Project

FutureGrid (FG) [13] provides computing capabilities that will enable researchers to tackle complex research challenges related to the use and security of grids and clouds. These include topics ranging from authentication, authorization, scheduling, virtualization, middleware design, interface design and cybersecurity, to the optimization of Grid-enabled and cloud-enabled computational schemes for researchers in astronomy, chemistry, biology, engineering, atmospheric science and epidemiology.

[1]Due to the restrictions in VMWare's licensing agreement, benchmark results are unavailable.

The test-bed includes a geographically distributed set of heterogeneous computing systems, a data management system that will hold both metadata and a growing library of software images necessary for cloud computing, and a dedicated network allowing isolated, secure experiments. The test-bed will support virtual machine-based environments, as well as operating systems on native hardware for experiments aimed at minimizing overhead and maximizing performance. The project partners will integrate existing open-source software packages to create an easy-to-use software environment that supports the instantiation, execution and recording of grid and cloud computing experiments.

One of the goals of the project is to understand the behavior and utility of cloud computing approaches. Recently, cloud computing has become quite popular and a multitude of cloud computing middleware have been developed. However, it is not clear at this time which of these toolkits will become the users' choice toolkit. FG provides the ability to compare these frameworks with each other while considering real scientific applications. Hence, researchers will be able to measure the overhead of cloud technology by requesting linked experiments on both virtual and bare-metal systems, providing them valuable information that will help them decide which infrastructure suits them better and also help users that want to transition from one environment to the other. These interests and research objectives make the FutureGrid project the perfect match for this work. Furthermore, its hoped that the results gleaned from these experiments will have a direct impact on the FutureGrid deployment itself.

B. Testing Environment

Currently, FutureGrid's premier supercomputer is the *India* system, a 256 CPU IBM iDataPlex machine consisting of 1024 cores, 2048 GB of ram, and 335 TB of storage within the Indiana University Data Center. In specirfic, each compute node of India has two Intel Xeon 5570 quad core CPUs running at 2.93Ghz, 24GB of DDR2 Ram, and dual data rate Infiniband 20Gbps. Four nodes in total allocated from India for these experiments. All were loaded with Red Hat Enterprise Linux server 5.5 x86_64 with the 2.6.18-194.8.1.el5 kernel patched. From the four nodes, three were installed with different hypervisors; Xen version 3.1, KVM (build 83), and VirtualBox 3.2.10, and the forth node was left as-is to act as a control for bare-metal performance. Each guest virtual machine was also built using Red Hat EL server 5.5 running an unmodified kernel using full virtualization techniques. Within each VM is each benchmark suite. All initial tests were conducted giving the guest VM 8 cores and 8Gb of ram to properly span a compute node. Each benchmark was run a total of 10 times, with the results averaged to ensure consistent results.

III. Performance Comparison

For the performance comparison of each virtual machine will be based on two well known industry standard performance benchmarks; HPCC and spec. These two benchmarks are well known for their standardized reproducible results in the HPC community. The National Science Foundation (NSF), Department of Energy (DOE) and DARPA are all sponsors of the HPCC benchmarks.

A. HPCC Benchmarks

The HPCC Benchmarks [14], [15]are an industry standard for performing benchmarks for HPC systems. The benchmarks are aimed at testing the system on multiple levels to test their performance. It consists of 7 different tests:

- *HPL* - The Linpack TPP benchmark which measures the floating point rate of execution for solving a linear system of equations. This benchmark is perhaps the most important benchmark within HPC today, as it is the basis of evaluation for the Top 500 list [16].
- DGEMM - measures the floating point rate of execution of double precision real matrix-matrix multiplication.
- *STREAM* - A simple synthetic benchmark program that measures sustainable memory bandwidth (in GB/s) and the corresponding computation rate for simple vector kernel.
- *PTRANS* - Parallel matrix transpose exercises the communications where pairs of processors communicate with each other simultaneously. It is a useful test of the total communications capacity of the network.
- *RandomAccess* - measures the rate of integer random updates of memory (GUPS).
- *FFT* - Measures the floating point rate of execution of double precision complex one-dimensional Discrete Fourier Transform (DFT).
- *Communication bandwidth and latency* - A set of tests to measure latency and bandwidth of a number of simultaneous communication patterns; based on b_eff (effective bandwidth benchmark).

B. SPEC Benchmarks

The Standard Performance Evaluation Corporation (SPEC) [17], [18] is the other major standard for evaluation of benchmarking systems. SPEC has several different testing components that can be utilized to benchmark a system. For our benchmarking comparison we will use:

- *CPU2006* - Evaluates the CPU on, comparing compute-intensive integer performance using CINT2006 and comparing compute-intensive floating point performance using CFP2006.
- *MPI2007* - Evaluates MPI-parallel, floating-point, compute-intensive performance. It can be used to compare different hardware ar- chitecture, network interconnects, processors, memory, compilers, and MPI implementations.
- *OMP2001* - Evaluates performance based on OpenMP which is the standard for shared-memory parallel processing.
- *VIRT2010* - Evaluates virtualization services, such as hardware, virtualized platform, virtualized guest operating system and application software, hardware virtualization, operating system virtualization, and hardware partitioning schemes.

These benchmarks provide a means to stress and compare processor, memory, inter-process communication, network, and overall performance and throughput of a system. These benchmarks are of extreme importance to the HPC community as they are often directly correlated with overall application performance [19].

IV. Discussion

We look to prepare and present detailed description and analysis of each benchmark conducted. This includes specific comparison and normalized comparisons between bare metal and each VM type, and our interpretation of the rationale for each benchmark result set. As a consequence, these hypervisors will be evaluated in an HPC setting on a number of different metrics to find the best possible performance. These results will be presented in a number of charts to easily compare each benchmark. It is our hope and intent that these results will be directly applicable to the FutureGrid project as well as other new Cloud deployments, leading to faster and more efficient Cloud platforms to solve the world's largest computational problems.

Acknowledgment

This document was developed with support from the National Science Foundation (NSF) under Grant No. 0910812 to Indiana University for "FutureGrid: An Experimental, High-Performance

Grid Test-bed." Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF. We would also like to personally thank Gregor von Laszewski, Greg Pike, Archit Kulshrestha, Joseph C Rinkovsky, Fugang Wang, and the rest of the FutureGrid team for their continued help and support.

REFERENCES

[1] L. Wang, G. von Laszewski, A. J. Younge, X. He, M. Kunze, J. Tao, and C. Fu, "Cloud Computing: a Perspective Study," *New Generation Computing*, vol. 28, no. 2, pp. 137 – 146, March 2010.

[2] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. L. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," in *Proceedings of the 19th ACM Symposium on Operating Systems Principles*, New York, U. S. A., Oct. 2003, pp. 164–177.

[3] K. Adams and O. Agesen, "A comparison of software and hardware techniques for x86 virtualization," in *Proceedings of the 12th international conference on Architectural support for programming languages and operating systems*. ACM, 2006, pp. 2–13, vMware.

[4] "Amazon elastic compute cloud," [Online], http://aws.amazon.com/ec2/.

[5] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, "The Eucalyptus Open-source Cloud-computing System," *Proceedings of Cloud Computing and Its Applications*, 2008.

[6] J. Fontan, T. Vazquez, L. Gonzalez, R. S. Montero, and I. M. Llorente, "OpenNEbula: The Open Source Virtual Machine Manager for Cluster Computing," in *Open Source Grid and Cluster Software Conference*, San Francisco, CA, USA, May 2008.

[7] S. Adabala, V. Chadha, P. Chawla, R. Figueiredo, J. Fortes, I. Krsul, A. Matsunaga, M. Tsugawa, J. Zhang, M. Zhao, L. Zhu, and X. Zhu, "From virtualized resources to virtual computing Grids: the In-VIGO system," *Future Generation Comp. Syst.*, vol. 21, no. 6, pp. 896–909, 2005.

[8] J. Chase, D. Irwin, L. Grit, J. Moore, and S. Sprenkle, "Dynamic virtual clusters in a grid site manager," in *12th IEEE International Symposium on High Performance Distributed Computing, 2003. Proceedings*, 2003, pp. 90–100.

[9] K. Keahey, I. Foster, T. Freeman, X. Zhang, and D. Galron, "Virtual Workspaces in the Grid," *Lecture Notes in Computer Science*, vol. 3648, pp. 421–431, 2005. [Online]. Available: http://workspace.globus.org/papers/VW_EuroPar05.pdf

[10] P. Padala, X. Zhu, Z. Wang, S. Singhal, and K. Shin, "Performance evaluation of virtualization technologies for server consolidation," HP Laboratories, Tech. Rep., 2007.

[11] J. Watson, "Virtualbox: bits and bytes masquerading as machines," *Linux Journal*, vol. 2008, no. 166, p. 1, 2008.

[12] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori, "kvm: the Linux virtual machine monitor," in *Proceedings of the Linux Symposium*, vol. 1, 2007, pp. 225–230.

[13] "Future grid web page," Web Page, 2009. [Online]. Available: http://www.futuregrid.org

[14] P. Luszczek, D. Bailey, J. Dongarra, J. Kepner, R. Lucas, R. Rabenseifner, and D. Takahashi, "The HPC Challenge (HPCC) benchmark suite," in *SC06 Conference Tutorial*. Citeseer, 2006.

[15] J. Dongarra and P. Luszczek, "Reducing the time to tune parallel dense linear algebra routines with partial execution and performance modelling," niversity of Tennessee Computer Science Technical Report, Tech. Rep., 2010.

[16] J. Dongarra, H. Meuer, and E. Strohmaier, "Top 500 supercomputers," website, November 2008.

[17] K. Dixit, "The SPEC benchmarks," *Parallel Computing*, vol. 17, no. 10-11, pp. 1195–1209, 1991.

[18] S. P. E. Corporation, "Spec's benchmarks and published results," Website, July 2010. [Online]. Available: http://www.spec.org/benchmarks.html

[19] J. J. Dujmovic and I. Dujmovic, "Evolution and evaluation of spec benchmarks," *SIGMETRICS Perform. Eval. Rev.*, vol. 26, no. 3, pp. 2–9, 1998.