 ISSN 2349-4506
Impact Factor: 2.265

Global Journal of Engineering Science and Research Management

VIRTUAL MACHINE ENVIRONMENT AND FRAMEWORK PERFORMANCE

Arya Yogendra Chouhan*[1] Gopal Sahu[2] Prakash Kumar Sen[3]
*[1]Student, Mechanical Engineering, Kirodimal Institute of Technology, Raigarh (C.G.).
[2]Faculty, Mechanical Engineering, Kirodimal Institute of Technology, Raigarh (C.G.).
[3]Faculty, Mechanical Engineering, Kirodimal Institute of Technology, Raigarh (C.G.).

KEYWORDS: Virtual machine, Virtualization, VMware.

ABSTRACT

The Virtual machines are an important component of modern portable environments. Their performance is critical to the success of such environments, but they are difficult to design well because they are subject to conflicting goals. On the one hand, they offer a way to hide the differences between instruction architectures; on the other, they must be implemented efficiently on a variety of underlying machines. A comparison of the engineering and evolution of the Inferno and Java virtual machines provides insight into the tradeoffs in their design and implementation. We argue that the design of virtual machines should be rooted in the nature of modern processors, not language interpreters, with an eye towards on-the-fly compilation rather than interpretation or special-purpose silicon.

INTRODUCTION

A virtual machine (VM) is a software implementation of a machine that executes programs like a physical machine; it is an emulation of a particular computer system. Virtual machines operate based on the computer architecture and functions of a real or hypothetical computer and their implementations may involve specialized hardware, software or a combination of both. Different virtualization techniques are used based on the desired usage. It also provides a complete substitute for the targeted real machine and a level of functionality required for the execution of a complete operating system [1]**.**

HISTORY

Virtual machines and process virtual machines both are date to the 1960s, and continue to be areas of active development. The first widely available virtual machine architecture was the CP-67/CMS. Although the industry uses diverse terms to describe these techniques, they are usually known as emulation, complete virtualization, Para virtualization, and operating system (OS)-level virtualization [2].

VIRTUAL MACHINES

Virtual machine (VM) technologies have become well-liked in both industry and academia due to various features designed to ease system management and administration. While a VM based environment can greatly. Popek and Goldberg have defined VM as "an efficient, isolated duplicate of a real machine". Presently virtual machines which have no directed interaction any of the real hardware are popular [3]. Virtual machine (VM) is software that emulates the physical machine. Virtual machines are categorized into two groups as Process Virtual Machine and System Virtual Machine [4].

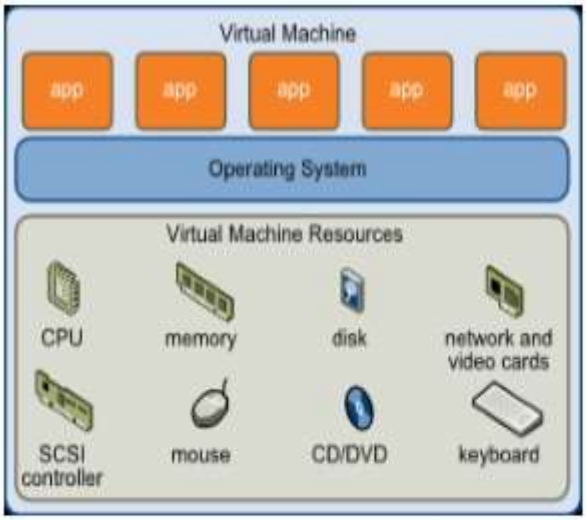

Figure3. 1: virtual machine.

PROCESS VIRTUAL MACHINES

A process virtual machine or application Virtual machine is designed to run a single program with a single process. It runs just like a regular application within the host OS as a process. It is created when process is initiated and destroyed when the process exits or dies. A Process VM is sometimes referred to as application virtual machine. This VM mainly aims at providing a Platform-independent development environment. Java programming language is platform Independent as it implements Java Virtual Machine (JVM) which is a process VM [5].

HOW VIRTUAL MACHINES WORK

VMware is a virtual-machine platform that makes it possible to run an unmodified operating system as a user-level application. The OS running within VMware Can is rebooted, crashed, modified, and reinstalled without affecting the integrity of other Applications running on the computer [6]. A virtual-machine monitor is an additional layer of Software between the hardware and the operating system that virtualizes all of the hardware resources of the machine. It essentially creates a virtual hardware execution environment called a "virtual machine" (VM). Multiple VMs can be used at the same time, and each VM provides isolation from the real hardware and other activities of the underlying system Because, it provides the illusion of standard PC (Personal Computer) hardware within a VM, VMware can be used to run multiple unmodified PC operating systems simultaneously on the same machine by running each operating system in its own VM. An OS running as a user-level application on top of VMware is called a "guest OS." The native OS originally running on the Real hardware is called the "host OS." VMware is low-level enough to make a guest OS appear to be receiving hardware interrupts And behave as if it were the only OS on the machine. At the same time, it provides isolation so that a failure in or misbehaving of a guest OS does not affect other guest OSs or the underlying system. For instance,

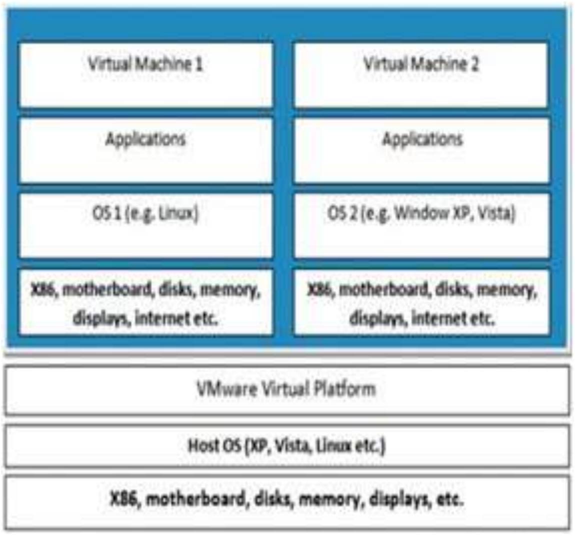

Fig.5.1: Two Virtual Machines Hosted on One Host Operating System

a guest OS crashing will not crash the underlying system. As opposed to a software simulator, much of the code running in a VM executes directly on the hardware without interpretation. Operating systems currently supported as guest operating systems under VMware include Windows 95/98/2000/NT, FreeBSD, Solaris, Novell Netware, DOS, and Linux, all of which run unmodified. Theoretically, any OS that can run on an x86 architecture can run as a guest OS, since it will see a complete virtualized PC environment. For host operating systems, VMware currently runs and is supported on Windows Vista, XP, 2000/NT and Linux [7].

FRAMEWORK

In this section we give a framework to analyze the performance of general virtual machines environment. The analytic queue network results are sometimes criticized for inaccurate performance evaluation and then it is invaluable during the development process. However, without analytic results, it is too cost for the designers to test every new design proposal using all kinds of experiments. Furthermore, the designer cannot get the performance guarantees for any new design. Consequently, it is important for the designer to yield some analytic results though there is a gap between practice and theory [8].

Global Journal of Engineering Science and Research Management

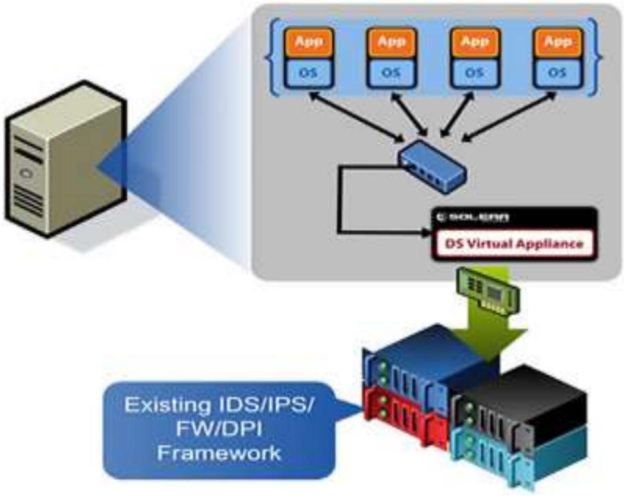

Fig.6.1: Framework of virtual machine process.

The virtualized system consists total *n* virtual machines above the VMM (or hypervisor). Each virtual machine has its own operating system. The detail of the framework is illustrated in Fig. 1. In the following, we will adopt the queue network theory to analyze the performance of the given framework. The framework consists of four components, input, workload analysis or monitor, service and performance analysis. In the input part, we must specify the queue network model. Then the workload in each virtual machine is described. In the case of single class, the workload is described as the work intensity for each resource. In the case of multiple classes, a matrix of work intensive for every resources is required [9].

PERFORMANCE OF VIRTUAL NETWORKS

Virtualization does provide an excellent flexibility and portability, but can also introduce degradation in network performance, especially in high performance throughput and low latency devices. This section analyzes the overhead associated with VMware-based virtual networks. The results from our experiments can be used as benchmarks and as reference for comparison testing [10].

CONCLUSIONS

Virtualization can create real world business environment as closely as possible in an academic Setting the performance study conducted in this paper, although on small scale, shows that there would be no significant performance overhead on a virtual network of host machines and virtual machines. In conclusion, virtualization is a new growing trend in the IT industry. Businesses as well as educational communities can equally be benefited from it despite the overhead involved in setting up a virtual network [11].

REFERENCES

1. https://en.wikipedia.org/wiki/Virtual machine
2. https://en.wikipedia.org/Historyof CP/CMS virtual machine
3. Smith, James; Nair, Ravi (2005). "The Architecture of Virtual Machines". Computer (IEEE Computer Society) 38 (5): 32–38. doi:10.1109/MC.2005.173.
4. journal paper of performance analysis of kernel-based virtual machine
5. https://en.wikipedia.org/wiki/ process Virtual machine.
6. http://www.vmware.com
7. journal paper of VIRTUAL MACHINES AND NETWORKS –INSTALLATION,PERFORMANCE, STUDY, Vol.3, No.1, January 2011
8. Fabr´ıcio Benevenuto1, C´esar Fernandes1, Matheus Santos1, Virg´ılio Almeida1,Jussara Almeida1, G.(John) Janakiraman2, Jos´e Renato Santos2 "Performance Models for Virtualized Applications". HP Labs.
9. M. Yehuda, J. D. Mason "The Xen Hypervisor and its IO Subsystem". IBM Haifa Research Lab, IBM Linux
10. http://www.netperf.org/netperf
11. research paper in Designs of Virtual file system in Cloud Environment by KL University, Guntur, Andhra Pradesh, India.