# ABSTRACT

In this project, we propose a scheme that manages the Performance and resource of virtual machines that are used to host computing applications. As the size and complexity of modern computing systems keep increasing to meet the demanding requirements of Performance applications, manageability is becoming an important concern to achieve both performance and productivity computing. Meanwhile, virtual machine (VM) technologies have become well-liked in both industry and academia due to various features designed to ease system management and administration. Virtualization allows computer users to utilize their resources more efficiently and effectively. Operating system that runs on top of the Virtual Machine or Hypervisor is called guest OS. The Virtual Machine is an abstraction of the real physical machine. We provide an analytic framework for the performance analyzing either without running a system or in a runnable real system. With these market trends toward virtual environments, many research groups are developing evaluation tools to check the performance of virtual systems and their overheads. However, the performance characterization in virtual environments has not been established yet for many challenging issues.The benefits of virtualization are typically considered to be server consolidation, (leading to the reduction of power and cooling costs) increased availability, isolation, ease of operating system deployment and simplified disaster recovery. One main challenge for virtualization, the need to maximize throughput with minimal loss of CPU and I/O efficiency. We are going to install various virtual softwares such as Xen , VMware  and analyze their performances based on their CPU utilization ,memory storage ,connection time, throughput ,CPU I/O wait and security issues.To represent these statistics we will be making use of various mathematical tools such as graphs ,histograms ,tables etc.

**Submitted by:**

AkritiKumari (15IT107)

Ankita Bhalavi(15IT108)

Supriya Sahoo(15IT145)

VandanaBaidya(15IT250)

## ACKNOWLEDGEMENT

This project would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them. We are highly indebted to the Department of Information Technology, NITK for their guidance and constant supervision as well as for providing necessary information and resources for the project and also for their support in completing the project. We would like to express our gratitude towards our teacher Ms. Shruti for her kind co-operation and encouragement which helped us in completion of this project. Our thanks and appreciations also go to our group members in developing the project and people who have willingly helped us out with their abilities.


Signature of course instructor — Signature of Examiner

(Ms. J.R. Shruti) — (Ms. J.R. Shruti)

# CONTENTS

# 1. INTRODUCTION

Performance analysis and troubleshooting of cloud applications are challenging. In particular, identifying the root causes of performance problems is quite difficult. This is because profiling tools based on processor performance counters do not yet work well for an entire virtualized environment, which is the underlying infrastructure in cloud computing. In this work, we explore an approach for unified performance profiling of an entire virtual environment by sampling only at the virtual machine monitor (VMM) level and applying common-time-based analysis across the entire virtual environment from a VMM to all guests on a host machine. Our approach involves three steps: centralized data sampling at VMM-level, generation of symbol map for running programs in guests, and unified analysis of the entire virtualized environment with common time by the host-time-axis. Finally, our results demonstrate accurate profiling. In addition, we achieved a lower overhead than in a previous study as a result of having no additional context switches by the virtual interrupt injection into the guest during measurement.

# 2. DESIGN AND IMPLEMENTATION

## 2.1 SYSTEM REQUIREMENTS:

### The system requirements for the XenServer are:

| | |
|---|---|
| **CPUs** | One or more 64-bit x86 CPU(s), 1.5GHz minimum, 2 GHz or faster multicore CPU recommended.<br>To support VMs running Windows, an Intel VT or AMD-V 64-bit x86-based system with one or more CPU(s) is required. |
| **RAM** | 2GB minimum, 4GB or more recommended |
| **Disk Space** | Locally attached storage (PATA, SATA, SCSI) with 16GB of disk space minimum, 60GB of disk space recommended, or SAN via HBA (not via software) if installing with multipath boot from SAN (See XenServer HCL for a detailed list of compatible storage solutions). Product installation creates two 4GB partitions for the XenServer host control domain. |
| **Network** | 100Mbit/s or faster NIC. One or more gigabit NIC(s) is recommended for faster P2V and export/import data transfers and VM live migration. |

### Windows 10 Installation Requirements for VirtualBox:

- **Processor** : 1GHz or faster with support for PAE, NX, and SSE2
- **RAM** : 1GB (32-bit) or 2GB (64-bit)
- **Hard disk space** : 16GB
- 64-bit x86 Intel Core 2 Duo Processor or equivalent, AMD Athlon 64 FX Dual Core Processor or equivalent.
- 1.3GHz or faster core speed.
- 2GB RAM minimum/ 4GB RAM recommended
-

## 2.2 IMPLEMENTATION:

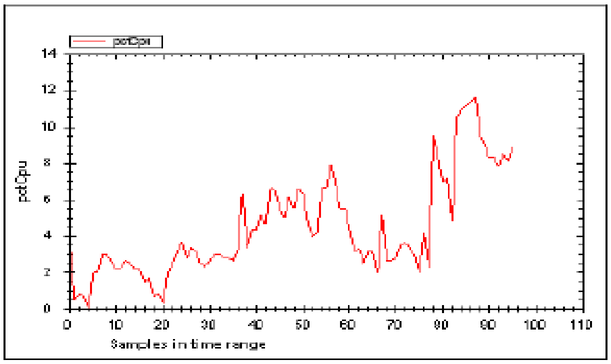
- Windows+R->resmon->perfmon

**memory usage commands on command prompt are:**

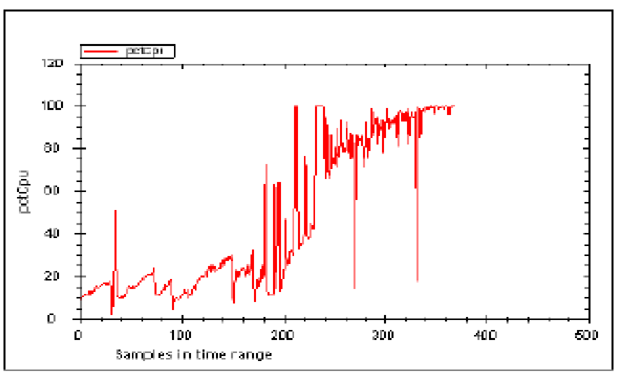
- systeminfo |find "Available Physical Memory"
- wmic memorychip get capacity
- typeperf "\Memory\Available Bytes"
- typeperf -qx "\Memory"
- wmic ComputerSystem get TotalPhysicalMemory
- wmic OS get FreePhysicalMemory
- wmic cpu get loadpercentage /format:value
- wmic os get freephysicalmemory /format:value
- wmic os get freevirtualmemory /format:value
- systeminfo | findstr Memory
- TASKLIST
- wmic memorychip get capacity
- PERFMON /RES
- DEFRAG /C /V /V
- perfmon /rel
- perfmon /report
- perfmon /res
- perfmon /sys
- perfmon /comp

# RESULTS

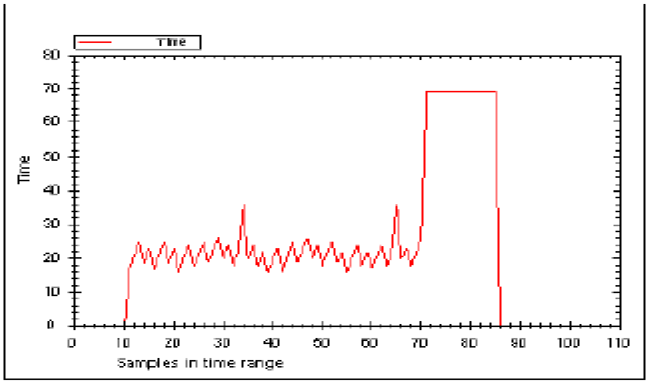
In order to demonstrate the causes of virtualization overhead on the Xen VMM, in the VM CPU utilization is slightly higher than the sum of the utilizations of the other CPUs.




**Graph for CPU Utilizations with Threshold Values**




**Graph for CPU Utilizations Beyond the Threshold Values**




**Graph  for Idle Time Utilizations**

In the performance analysis of the vm, threads that are scheduled by a virtual machine instead of natively by the underlying operating system. Threads emulate multithreaded environments without relying on any native OS capabilities, and they are manageable in user space instead of kernel space ,enabling them to work in environments that do not have native thread support.

# CONCLUSION

We have presented a performance measurement study focused on network I/O application, one of the dominating workloads in cloud-based virtual data centers .We first show that current implementation of virtual machine monitor does not provide sufficient performance isolation to guarantee the effectiveness of resource sharing across multiple virtual machine instances running on a single physical host machine, especially when applications running on neighboring VMs are competing for computing and communication resources. Then we present the detailed analysis on different factors that can impact the throughput performance and resource sharing effectiveness.

Concretely, we presented our measurement study and analysis along three dimensions: 1) the performance impact of idle instances on applications that are running concurrently on the same physical host; 2) the performance impact of colocating applications in a virtusalized data center; and 3) how different CPU resource scheduling and allocation strategies and different workload rates may impact the performance of a virtualized system.

## REFERENCES


- Deshi Ye, Q. He, Hua Chen, Jianhua Che “A framework to evaluate and predict performances in virtual machines Environment”. 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing
- https://en.wikipedia.org/wiki/Virtualization
- *Graziano, Charles. "A performance analysis of Xen and KVM hypervisors for hosting the Xen Worlds Project"*