ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Analysis of Net I/O Performance By Using Enhanced CBIR Systems

M.Archana
Assistant Professor,
Depatment of CSE,
TKR College of Engineering & Technology.

A.Roja
UG Scholar,
Depatment of CSE,
TKR College of Engineering & Technology.

B.Soundarya
UG Scholar,
Depatment of CSE,
TKR College of Engineering & Technology.

D.Sneha
UG Scholar,
Depatment of CSE,
TKR College of Engineering & Technology.

Abstract:

Quality of service (QoS) is the overall performance of a telephony or computer network, particularly the performance seen by the users of the network.n the services domain, end-to-end Quality of Service has also been discussed in the case of composite services (consisting of atomic services) or applications (consisting of application components).

Moreover, in cloud computing end-to-end QoS has been the focus of various research efforts aiming at the provision of QoS guarantees across the cloud service models.Cloud computing is a kind of grid computing; it has evolved by addressing the QoS (quality of service) and reliability problems.

Cloud computing provides the tools and technologies to build data/compute intensive parallel applications with much more affordable prices compared to traditional parallel computing techniques. In this paper we are going to measure, quantify and experiment on QoS Indicator in Cloud: Parallel processing of CPU-intensive and network-intensive workloads on VMs. A system virtual machine provides a complete system platform which supports the execution of a complete operating system (OS).

These usually emulate an existing architecture, and are built with the purpose of either providing a platform to run programs where the real hardware is not available for use (for example, executing on otherwise obsolete platforms), or of having multiple instances of virtual machines leading to more efficient use of computing resources, both in terms of energy consumption and cost effectiveness (known as hardware virtualization, the key to a cloud computing environment), or both.

Keywords:

Virtual machines, QoS, Cloud Computing, Networks, bandwidth.

Introduction:

Cloud computing is a computing term or metaphor that evolved in the late 1900s, based on utility and consumption of computer resources. Cloud computing involves deploying groups of remote servers and software networks that allow different kinds of data sources be uploaded for real time processing to generate computing results without the need to store processed data on the cloud. Clouds can be classified as public, private or hybrid.The main enabling technology for cloud computing is virtualization. Virtualization software separates a physical computing device into one or more “virtual” devices, each of which can be easily used and managed to perform computing tasks.

With operating system–level virtualization essentially creating a scalable system of multiple independent computing devices, idle computing resources can be allocated and used more efficiently. Virtualization provides the agility required to speed up IT operations, and reduces cost by increasing infrastructure utilization. Autonomic computing automates the process through which the user can provision resources on-demand.

By minimizing user involvement, automation speeds up the process, reduces labor costs and reduces the possibility of human errors. Users routinely face difficult business problems. Cloud computing adopts concepts from Service-oriented Architecture (SOA) that can help the user break these problems into services that can be integrated to provide a solution.

Cloud computing provides all of its resources as services, and makes use of the well-established standards and best practices gained in the domain of SOA to allow global and easy access to cloud services in a standardized way. Cloud computing also leverages concepts from utility computing to provide metrics for the services used. Such metrics are at the core of the public cloud pay-per-use models. In addition, measured services are an essential part of the feedback loop in autonomic computing, allowing services to scale on-demand and to perform automatic failure recovery.In computing, a virtual machine (VM) 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.Classification of virtual machines can be based on the degree to which they implement functionality of targeted real machines. That way, system virtual machines (also known as full virtualization VMs) provide a complete substitute for the targeted real machine and a level of functionality required for the execution of a complete operating system. On the other hand, process virtual machines are designed to execute a single computer program by providing an abstracted and platform-independent program execution environment.

Existing System :

Virtualization technology offers many advantages in cloud-based data centers, such as reducing total costs of ownership, improving energy efficiency, and resource utilization. By providing physical resources sharing, fault isolation and live migration, virtualization allows diverse applications to run in isolated environments through creating multiple virtual machines (VMs) on shared hardware platforms, and manage resource sharing across VMs via virtual machine monitor (VMM). Although VMM has the ability to slice resources and allocate the shares to different VMs, our measurement study shows that the performance of applications running in one VM would be affected by the applications running on its neighbor VMs, especially when these VMs are running at high rates of network I/O workloads. More importantly, our study shows that the level of performance interference mainly depends on the degree of the competition that the concurrent applications running in separate VMs may have in terms of shared resources.

We argue that the in-depth understanding of potential interference factors among VMs running on a shared hardware platform is critical for effective management in virtualized clouds, and an open challenge in current virtualization research and deployment.

In this paper, we study performance interference among multiple VMs running on the same hardware platform with the focus on network I/O processing. The main motivation for targeting our measurement study on performance interference of processing concurrent network I/O workloads is simply because network I/O applications are becoming dominating workloads in most cloud-based data centers now.

Disadvantages:

By carefully designing of measurement tested and the set of performance metrics, we derive some important factors of I/O performance conflicts based on application throughput interference and net I/O interference. Our performance measurements and analyses also provide some insights into performance optimizations for CPU scheduler and I/O channel, and as well efficiency management of workloads and VM configurations, such as CPU assignment across VMs.

Proposed System :

To address the problem of I/O efficiency in the driver domain model, Mansley proposed “direct I/O” to alleviate the pressure in driver domain by allowing guest domains to access hardware directly. Though the virtualized system can act as the native system in some cases, direct I/O is not considered in our work, as it lacks of dedicated driver domain to perform fault isolation, which is considered to be the most important function for system virtualization. Implemented to monitor the detailed system-level values of each VM, such as CPU usage, I/O count, and execution count.

Based on the monitoring results, Share Guard and SEDF-DC mechanisms are proposed to improve performance isolation studied the effects of performance interference between two VMs hosted on the same physical platform by collecting the runtime performance characteristics of different types of concrete applications.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Advantages :

• A similar but reverse procedure is applied to send a packet on the send path (TX), except that no explicit memory page exchange is involved. Only the ownership of physical page is transferred instead of the real page.

• Netback copies the received data to granted page in guest domain. Finally, guest domain receives the packet as if it comes directly from NIC.

Modules Description :

Xen I/O Overview :

Xen is a popular open-source x86 VMM, supporting both full-virtualization and para-virtualization. Xen uses para-virtualization as a more efficient and lower overhead mode of virtualizations. In para-virtualization I/O mode, Xen VMM layer uses asynchronoushypercall-mechanism to deliver virtual interrupts and other notifications among domains via event channel.

I/O Workloads :

For each of the three alternative hardware platforms in Table 1, we first evaluate the actual performance results and characteristics of each I/O workload running in single guest domain, which serves asbasecasein the rest of this paper.

We only discuss the experimental details ofbasecase on Platform I, a representative physical host for performance interference analysis in next two sections. Throughput Performance Interference This section focuses on studying interference of running multiple net I/O workloads in isolated VMs. We first show the performance interference on Platforms I and II, and then provide analysis of performance interference in each of the two setups, respectively, focusing on understanding the impacts of running different workloads in conjunction on the aggregate throughput.

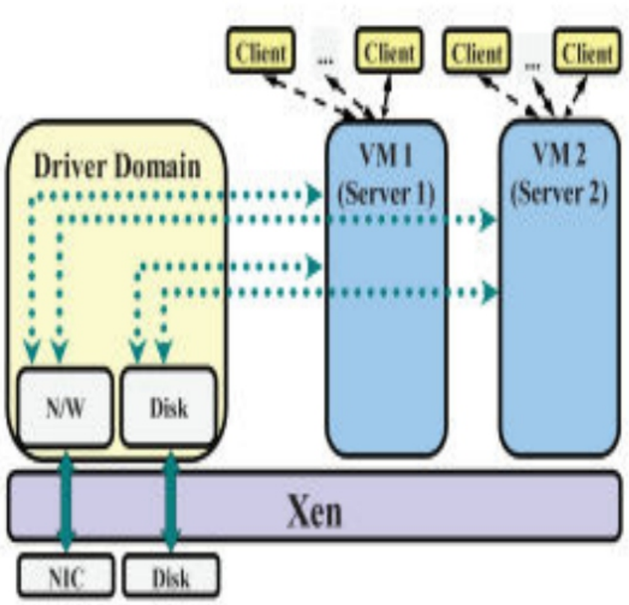
I/O Performance Interference :

From our throughput performance interference analysis in previous section, we know that the combination of both CPU-bound workloads for Platforms causes throughput interference mainly due to Domo demanding for fast I/O processing and grant mechanism management, while the combination of both network intensive workloads for Platform I and incurs the highest VMM switch overhead.

Multicore Performance Interference :

We dedicate this section to analyze the impact of multicore and larger L2 cache size on performance interference of co-locating CPU-bound or network-bound workloads in a virtualized cloud environment. First, the experimental setup is introduced.

Flow Diagram:



ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

CONCLUSIONS :

We presented an extensive experimental study of performance interference in running CPU-bound and networkbound workloads on separate VMs hosted by the same physical infrastructure, enabled by Xen VMM. Eight metrics are used to analyze the performance interference among VMs serving net I/O workloads that are either CPU bound or network bound. Based on measurements and observations, we conclude with five key findings that are critical to effective management of virtualized cloud for both cloud service providers and consumers. First, running networkbound workloads in isolated VMs on a shared hardware platform leads to high overheads due to extensive context switches and events in driver domain and VMM.

Second, colocating CPU-bound workloads in isolated VMs on a shared hardware platform incurs high-CPU contention due to the demand for fast memory page exchanges in I/O channel. Third, running CPU-bound and network-bound workloads in conjunction incurs the least resource contention, delivering higher aggregate performance. However, default credit scheduler treats network-bound workload unfairly under SMP system. Fourth, the performance of network-bound workload is not sensitive to CPU core assignment among VMs. In contrast, more cores pinned on driver domain delivers worse performance for CPU-bound workload.

Finally, due to fast I/O processing in I/O channel, the limited size on grant table can be a potential bottleneck in current Xen system. Identifying factors that impact total demand of exchanged memory pages is critical to in-depth understanding of interference overheads in driver domain and VMM layer.

REFERENCES :

[1] P. Apparao, S. Makineni, and D. Newell, "Characterization of Network Processing Overheads in Xen,"Proc. Second Int'l Workshop Virtualization Technology in Distributed Computing (VTDC '06), pp. 2-9, Nov. 2006.

[2] P. Barham, B. Dragovic, K.A. Fraser, S. Hand, T. Harris, A. Ho, E. Kotsovinos, A. Madhavapeddy, R. Neugebauer, I. Pratt, and A. Warfield, "Xen 2002," Technical Report UCAM-CL-TR-553, Univ. of Cambridge, Jan. 2003.

[3] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the Art of Virtualization,"Proc. Ninth ACM Symp. Operating ystems Principles (SOSP '03),pp. 164-177, Oct. 2003.

[4] L. Cherkasova and R. Gardner, "Measuring CPU Overhead for I/O Processing in the Xen Virtual Machine Monitor,"Proc. USENIX Ann. Technical Conf. (ATC '05),pp. 387-390, Apr. 2005.

[5] L. Cherkasova, D. Gupta, and A. Vahdat, "Comparison of the Three CPU Schedulers in Xen," Proc. ACM SIGMETRICS Performance Evaluation Rev. (PER '07),pp. 42-51, Sept. 2007.

[6] K. Fraser, S. Hand, R. Neugebauer, I. Pratt, A. Warfield, and M. Williamson, "Reconstructing I/O," Technical Report UCAM-CLTR-596, Univ. of Cambridge, Aug. 2004.

[7] K. Fraser, S. Hand, R. Neugebauer, I. Pratt, A. Warfield, and M. Williams, "Safe Hardware Access with the Xen Virtual Machine Monitor,"Proc. Operating System and Architectural Support for the on Demand IT Infrastructure (OASIS '04),pp. 1-10, Oct. 2004.

[8] D. Gupta, R. Gardner, and L. Cherkasova, "XenMon: QoS Monitoring and PerformanceProfiling Tool," Technical Report HPL-2005-187, HP Laboratories, Oct. 2005.

[9] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat, "Enforcing Performance Isolation across Virtual Machines in Xen,"Proc. ACM/IFIP/USENIX Seventh Int'l Middleware Conf. (Middleware '06), pp. 342-362, Nov. 2006.

[10] T. Hatori and H. Oi, "Implementation and Analysis of Large Receive Offload in a Virtualized System,"Proc. Virtualization Performance: Analysis, Characterization, and Tools (VPACT '08),Apr. 2008.