INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 9, Number 2, Pages 351–370

PERFORMANCE COMBINATIVE EVALUATION FROM SINGLE VIRTUAL MACHINE TO MULTIPLE VIRTUAL MACHINE SYSTEMS

KEJIANG YE, JIANHUA CHE, QINMING HE, DAWEI HUANG, AND XIAOHONG JIANG

Abstract. Virtualization technology is widely used in server consolidation, high performance computing, and cloud data center due to its benefits on high resource utilization, flexible manageability, and dynamically scalability. However, it also introduces additional performance overheads. It's worthy to evaluate the overheads and to find the bottleneck of virtualization in different scenarios. In this paper, we propose a combinative evaluation method to analyze the performance from single virtual machine to multiple virtual machine systems that measures and analyzes both the macro-performance and micro-performance. By correlating the analysis results of two-granularity performance data, some potential performance bottlenecks come out.

Key words. Virtualization, Performance, VMM, HPC, Virtual cluster.

1. Introduction

Virtualization technology has recently become increasing popular in wide areas, such as server consolidation [1, 2], high performance computing (HPC) [3, 4], and modern cloud data center [5, 6]. Single virtual machine system virtualized on single physical machine is a basic form. While multiple virtual machine system virtualized on single or multiple physical machines is a more common scenario (see Figure 1). It is obvious that virtualization brings many benefits such as flexible resource management, high reliability, performance isolation, and OS customization. In a typical virtualization system, resource virtualization of underlying hardware and concurrent execution of virtual machines are in the charge of virtual machine monitor (VMM) or hypervisor [7]. By creating the same view of underlying hardware and platform APIs from different vendors, virtual machine monitor enables virtual machines to run on any available physical machines. However, virtual machine monitor also complicates the implementation of traditional computer architectures and depresses the performance of some specific operations, so it's significant to assess the performance of virtualization in both single virtual machine and multiple virtual machine system.

Currently, there are several mature virtualization solutions (typical virtual machine monitors) for x86 system using different implementation methods, such VMware [8] and KVM [9] for full virtualization, Xen [10] for both para-virtualization and full virtualization, and OpenVZ [11] for operating system level virtualization. Different virtualization solutions have different strengths and weaknesses. For example, full virtualization supports both Linux and Windows virtual machines but the performance is relatively poor, while para-virtualization cannot support Windows virtual machine but the performance is better. It is necessary to choose a most appropriate virtualization method for particular purposes. Further, in the multiple virtual machine system, such as HPC virtualization system, different forms of communication

Received by the editors October 12, 2009 and, in revised form, April 28, 2010.
This research was supported by the National 973 Basic Research Program of China under grant NO.2007CB310900 and National Natural Science Foundation of China under grant NO. 60970125.

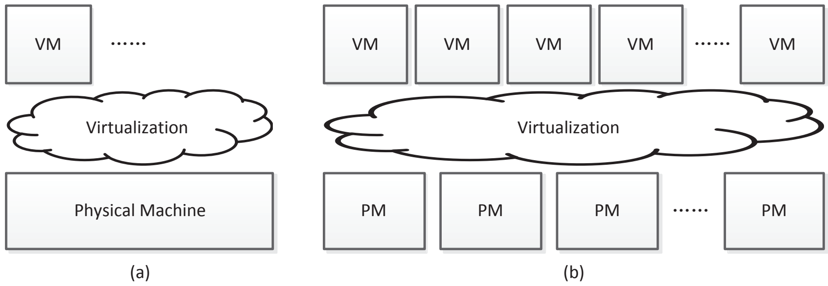

FIGURE 1. Typical virtualization scenarios: (a) the single virtual machine system on single physical machine; (b) the multiple virtual machine system on single or multiple physical machines.

between virtual machines is a major performance bottleneck, such as MPI (Message Passing Interface) communication, network communication, etc. How can the virtualization affect the communication efficiency is an important issue. Based on the above analysis, it is essential to analyze the performance overheads of single virtual machine system (such as processor virtualization, memory virtualization, disk I/O virtualization, network virtualization, etc.), compare the virtualization efficiency of different typical virtual machine monitors, and investigate the various communication overheads in multiple virtual machine system.

The performance of applications running in virtual machines is different from that in the native environment due to the existing of virtual machine monitor. A lot of work has been done in the performance evaluation on single virtual machine system focusing on the performance of CPU, memory, disk I/O, and network [10, 12–15], and the performance issues of server consolidation [1, 2, 16]. However, to our knowledge, there is still relatively few work focus on the performance evaluation of multiple virtual machine system, especially the HPC virtualization system. Some of the researchers have investigated into virtualization performance of HPC applications [4, 17–19]. However they didn't perform a deep analysis on the performance overheads and bottleneck of the network I/O processing mechanism in multiple virtual machine system. What's more, they only focused on the macro performance evaluation and didn't refer to micro performance analysis such as profiling analysis from the hardware architecture perspective.

In this paper, we firstly study the component virtualization overheads of single virtual machine system (such as the virtualization efficiency of processor, memory, disk I/O, network I/O, etc) by comparing the performance of different virtualization technologies, i.e. para-virtualization, full virtualization, and operating system level virtualization. Then we create two 16-node virtual clusters, and do a comprehensive performance evaluation of multiple virtual machine system (i.e. HPC virtualization system) to investigate the virtualization efficiency for HPC applications, including floating point computing performance, memory bandwidth performance, data transfer rate, network bandwidth and latency. Especially, we present a detailed assessment of various communication overheads with HPC applications running in virtual cluster. Besides, we also investigate the micro performance for both single virtual machine and multiple virtual machine systems. For single virtual machine system, we investigate some specific operations such as *system call* and *context switch*. While for multiple virtual machine system, we analyze the profiling data