Journal of ELECTRICAL ENGINEERING, VOL. 64, NO. 5, 2013, 272–282

PERFORMANCE EVALUATION OF VIRTUALIZATION TECHNIQUES FOR CONTROL AND ACCESS OF STORAGE SYSTEMS IN DATA CENTER APPLICATIONS

Mohammad Reza Ahmadi *

Virtualization is a new technology that creates virtual environments based on the existing physical resources. This article evaluates effect of virtualization techniques on control servers and access method in storage systems [1, 2]. In control server virtualization, we have presented a tile based evaluation based on heterogeneous workloads to compare several key parameters and demonstrate effectiveness of virtualization techniques. Moreover, we have evaluated the virtualized model using VMotion techniques and maximum consolidation. In access method, we have prepared three different scenarios using direct, semi-virtual, and virtual attachment models. We have evaluated the proposed models with several workloads including OLTP database, data streaming, file server, web server, *etc.* Results of evaluation for different criteria confirm that server virtualization technique has high throughput and CPU usage as well as good performance with noticeable agility. Also virtual technique is a successful alternative for accessing to the storage systems especially in large capacity systems. This technique can therefore be an effective solution for expansion of storage area and reduction of access time. Results of different evaluation and measurements demonstrate that the virtualization in control server and full virtual access provide better performance and more agility as well as more utilization in the systems and improve business continuity plan.

K e y w o r d s: virtual server, virtual access, storage system, workload

1 INTRODUCTION

Virtualization is the process of presenting a logical grouping or subset of computing resources so that they can be accessed in ways that give benefits over the original configuration. It is the most significant progress since the microprocessor introduction, in providing information and business secure systems [3]. The idea of virtualization can be used for most of the components in IT infrastructure such as networks, storages, servers, operating systems and applications. In server virtualization, the virtual environment allows for creation of multiple independent occurrence of an operating environment (logical or virtual servers) to run on a single physical server [4].

Virtualization allows physical servers to be carved into multiple virtual machines, and enabling a virtualized data center where applications hosts and managed in their dedicated virtual environment. Storage virtualization implements a virtual layer on top of the physical storages so that all physical devices hid from clients and prepare a pooling environment for using virtual spaces [5]. Thus, storage virtualization allows a system to execute its I/O operations without knowing exactly where or how the data are stored. In this way, virtualization provides larger virtual storage capacity, higher I/O performance and better data availability [6]. To show the role of virtualization technique in storage systems, we have compared performance of the three following access methods, Direct Attach Storage (Disk), Semi-Virtual [RAID (Redundant Array of Independent Disks) Array], and Virtual Storage system. Direct attach storage is a non-virtualized method where the storage is directly connected to control system. RAID is a semi virtualization technique that provides a virtual and independent area for redundancy purposes. Virtual storage prepares a set of logically separated virtual environments which are managed by a control system [7]. We have used the IOmeter software (date 10.05.2003 version) for performance evaluation of native and virtual storage systems. In the IOmeter assessment, we consider different workloads (*eg* file server, web server) with variable parameters (*eg* block size of I/O operations, percentage of read/write operations, and degree of randomness) [8]. Workload parameters are estimated by benchmark software, *ie* Netbench (a benchmark for different workloads). We have compared the results of evaluation for each workload based on the pre-defined parameters. The main goal of this arrangement is preparing an implementation platform and creating an assessment model for evaluating the efficiency and performance of the three proposed access methods. In addition, we have planned to show the improvement level in data transition rate and transaction time in different techniques. It this evaluation we have tested various technical specification of the three methods and compared the advantage and disadvantage of the alternative technologies in three scenarios [9]. The rest of this paper is organized as follows: Related work is the subject of Section 2. Section 3 presents control server and storage system architecture. Test bed architecture is described in Section 4. Section 5, present the system evaluation. Finally, we summarize the paper in Section 6.

* Research Institute of ICT (CSRI) Tehran, Iran, m.ahmadi@itrc.ac.ir

DOI: 10.2478/jee-2013-0040, Print ISSN 1335-3632, On-line ISSN 1339-309X,

2 RELATED WORK

There have been a number of available references in workload characterization which consider the behaviour of datacenter workloads [10, 11]. Those research works primarily focus on workload characteristics using synthetic tools such as IOmeter to model the applications in a well- known technique [12, 13]. Some operating systems have a single-host-version volume manager such as the logical volume manager (LVM) in Linux operating system [14]. These versions are inadequate for sharing the storage devices. RAID volume is a system of multiple hard drives for sharing or replicating data. RAID-5 (Disks striped disks with parity) combines three or more disks in a way that protects data against loss of any physical disk. The storage capacity of an array is reduced by disk numbers. RAID and LVM can be employed to construct a storage system for the global datacenter which needs a very reliable and large capacity storage environment [15]. On the other word, there have been a number of research works in server virtualization and its performance evaluation. Those researches primarily focus on evaluation of virtualization overhead. A performance evaluation methodology in virtual environments is presented in [16]. This paper considers four virtualization models for characterizing the performance of virtual servers. Performance evaluation in server consolidation using virtual machines is presented in [17, 18]. This paper evaluates the virtual machine overhead and the virtualization overhead for heavy database server running in virtual machines (VMs). In other work, VMmark benchmark of VMware corporations for performance evaluation of virtualized against non-virtualized servers over VMware ESX environments for IDC applications [16, 21]. These papers provide server virtualization in autonomic management of heterogeneous workloads. They explore in particular the use of server virtualization technology in the autonomic management of data centers running a heterogeneous mix of workloads [22]. In general, IT software working groups and vendors such as Parallels, VMware, XenSource and so on are developing various performance benchmarking tools to measure the performance of virtualized systems [23]. We have presented a technical evaluation of server virtualization and access model to validate the effect of virtualization technique.

3 CONTROL SERVER & STORAGE SYSTEM ARCHITECTURE

In order to introduce our system architecture, we have considered control server and three different access models, as it is shown in Fig. 1. The server system controls the storage modules and storage system has connection with storage system through direct, semi-virtual and virtual interface modules. To evaluate the performance of the virtualization technique and different interface modules, we have considered four different scenarios.

1. Control Server Scenario: virtualization in control server changes the server model architecture and establish a new layer called hypervisor between physical layer and operating systems. This layer includes several blocks of activities which are controlled by a management system. Hypervisor is a software environment that supports communication between hardware system and virtualization main activities such as memory, I/O, CPU, File system, network and resource management blocks. On top of this environment, a virtual server pool including variety of operating systems in different virtual machines can support variety of application software.

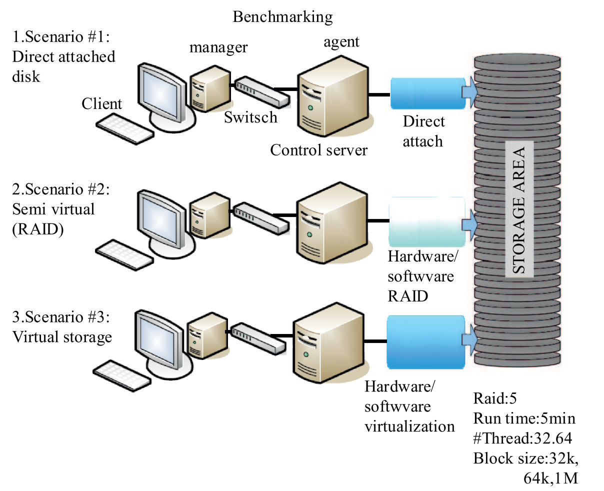

Fig. 1. Architecture of control server and storage access model

2. Direct Access Scenario: Direct attached model is the simplest type for accessing the control server to the storage systems and has direct connection with server. This is a hardware dependent model with weak and scattered management system. In this method, the coefficient of access to information is low, so any failure may cause disruption in connection. Moreover, the available storage capacity is limited by bus addressing in the control server.

3. Semi-Virtual Access Scenario: Semi Virtual storage creates an integrated unit of several hard disks. So, RAID systems can see all available disks as a source unit and can manage with different efficiency and security. RAID system provides facilities such as speed, backup and recovery technique based on selected mode such as RAID-0, RAID-1, RAID-5, and so on.

4. Virtual Access Scenario: Virtual Storage is a unique architecture that an operating system sees the storage units as a local attachment. Virtual Storages are implemented using appropriate virtualization hardware and software. It is a potential to have the most benefit of storage space, since all capacity are available in a shared storage environment. In this scenario the virtual environment can namely exceed from existing physical storage area.

Figure 1 shows the implementation model of server and three scenarios in our prototype system. As it is showed in this figure, the test systems are implemented in control

Table 1. Workloads block size percentage for file server and web server

Block Size	File Server Block Size Percentage	Web Server Block Size Percentage
512	10	22
1024	5	15
2048	5	8
4096	60	23
8192	2	15
16384	4	2
32768	4	6
65536	10	7
131072	0	1
524288	0	1

Table 2. Examplels of throughput metric

System	Throughput Metric
Disk & Storage	I/Os per Second
OLTP System	Transactions per Second
Web Server	Response Time per Second
File Server	MB transferred per Second

servers (Benchmarking Agent) and Clients (Benchmarking Manager). Here, Benchmarking is the I/O meter software and I/O Meter agent which are installed on the control server, and the I/O meter Manager which is installed on Client for all access scenarios. There are four workers in the I/O Meter software and we have considered 64 numbers of threads for each worker. The clients are connected to the Test Servers through a switching system. Test Servers are connected to the Storage Area through appropriate hardware interfaces. Storage Area is set of composed hard disks that forms high volume of capacity and all equipments in three scenarios are connected to the storage area. Test process is done through the Benchmarking Manager, as it sends the requests in a specific instruction to the Benchmarking Agent. The Benchmarking Agent, based on requested instructions will execute the test process and returns the result to the Benchmark Manager.

4 TEST BED ARCHITECTURE

In this section, we have introduced our test bed environment and workload modeling that we have used for control system and storage access.

4.1 CONTROL SERVER

a. *Hardware Environment*. In our test environment, we have used HP BLAD-45p server with full hardware resources which are connected to an HP storage system.

b. *Software Model*. In virtualization software, a group of virtual machines, virtual management module and virtual monitor creates software architecture.

c. *Platform and Software Tools*. In order to implement a standard model for evaluation, a list of necessary software and tools are given in appendix A. It presents a list of software which are necessary for implementation and evaluation of virtualization test plan. Selected software includes virtualization software, operating systems, workloads, database software, evaluation software, and monitoring tools.

d. *Workload Model*. In TILE method, system applications are installed in virtual machines. Virtual machines consist of a set of six standard servers in their testing environments. Consolidate Stack Units or a Tile includes file system, database, Java software, e-mail software and Web-based software which can run simultaneously in separate virtual machines. A significant tile in the integrated benchmark are based on a set of common datacenter workloads such as e-mail server, Java server, standby server, web server, database server and file server services.

4.2 ACCESS CONTROL

In this section, we have introduced the test environment and workload modelling that we have used for storage virtualization experimental test.

a. *Test Environment*. In our test environment, we have used the control server with FC connection cards that have connection to the SAN storage and a RAID controller in a storage system. In this experiment, we have installed the Dbench and IOmeter software and selected different workloads which are more popular in datacenter applications.

b. *Workloads Modelling*. Using fictitious working loads in IOmeter for applying to the applications is a well-known technique. IOmeter will determine the data of an I/O workload as a function of different variables: Reading percentage, percentage of rand, and block size. The specification of each workload has been extracted from technical documents. Workloads specification are: block size, the percentage of reading and writing, random percentage, the value of delay, the amount of burst workload, tuning of queue, and amount of response time. The Workload rates are 2k, 4k, and 8k. In OLTP (On Line Transaction Processing) it is using 100 % of size with 67 % reads, 100 % random and the block sizes of 2048, 4096, 8192. Workload specification for a file Server is 80 % read, 100 % random with zero delay. Also, workload specification for the web server is 100 % read, 100 % random with zero delay. In IOmeter software, the block size and percentage value of different workloads in file server and web server are shown in Table 1.

c. *Metrics*. Set X is called a *Metric Space* if for any two points p, q in X, the real value $\mathrm{d}(p,q)$ defines a distance between p and q as follows

$$\mathrm{d}(p,q) > 0 \text{ if } p \neq q \text{ and } \mathrm{d}(p,p) = 0\,, \tag{1}$$

$$\mathrm{d}(p,q) = \mathrm{d}(q,p)\,, \tag{2}$$

$$\text{for any } r \in X\ \mathrm{d}(p,q) \leq \mathrm{d}(p,r) + \mathrm{d}(r,q)\,. \tag{3}$$

Any function with the three above properties is called a metric.

d. *Definitions*.

1) *Throughput* is the average rate of successful message delivery over a communication channel or a computer system. The throughput measurement method is the number of operations per time-unit. Throughput depends on the nature of system and its operation. Examples of systems and throughput metrics are listed in Table 2.

When we consider the throughput as a metric, we should make sure that the desired operation is well defined. For a throughput, we can consider the following equation

$$\textit{Throughput} = \ \text{Min}\{\ \textit{Entered Loads, Server Capacity.}\} \quad (4)$$

Throughput increases almost linearly in low loads rate and saturates in a maximum level; this condition will happen when the network resources running in maximum efficiency. In some cases, for the loads which are more than saturation level, the throughput stagnates in a certain level or even decreases continuously. This phenomenon is called thrashing process. The reason for thrashing is that many processes are competing for a certain amount of main memory and spend lots of their CPU time and I/O for the error report rather than attention to the workload. In thrashing mode, operating system spends continuous time for additional overhead activities (because of increasing the load) and reduces the time intervals which are assigned for the CPU processes.

2) *Iops*. IOps is the average number of Input/Output action per second in the whole process of test. Read IOps is the mean of reading operations and write IOps is the mean of writing operations. IOps is an important benchmark for hard disks and other computer storage media which illustrates the system performance. Also, IOps is a permittivity metric for a disk that is measured by I/O per second. On the other hand, the Mbps (Megabits per second) is the average number of read and writes per second in the whole process of test. Mbps stands a Megabits per Seconds and equivalent to $1024 * 1024$ KB per seconds. Mbps indicates the bandwidth size (information flow in time unit) in a network or a media, that shows data transfer rate and demonstrates the efficiency of the network or media. Read Mbps is average number of only read Megabits per second and write Mbps is average number of only write Megabits per second.

3) *Average Transaction Time* (ATT). ATT means the average executing time of each transaction in the system. In other words, the average time between the starting time of a request and completing answer time. If there is no response for a process, the average transaction time is equal to average response time. The value of average transaction time for each system is very important and directly affect on the system performance. Whatsoever the average transaction time becomes smaller in a system, it more improves the system performance.

4) *Average I/O Response Time* (ART). ART is the average response time of each transaction or represents the average time which completes a transaction. The average I/O response time is the mean time between start and completion of each I/O operations during the whole process of a test in milliseconds. Average Read I/O Response Time is the mean of time between start and completion of each read operation. Average Write I/O Response Time is the mean of time between start and completion of each write operation.

5) *CPU Utilization Percentage*. CPU utilization is the processor time that just is used for executive threads not for the idle threads. In other words, CPU utilization is the time which spends for running the useful jobs. CPU utilization rate indicates the processor performance, and is one of the important factors for evaluating the performance of storage systems [12, 13].

6) *Tile Score value*. The scoring of VMmark Benchmark is computed after a valid run by the software. In computation, metrics of the workloads within each tile are computed and aggregated into a score of tile. This aggregation is performed by first normalizing the different performance metrics (such as MB/s and database commits/s) with respect to a reference system. Then a geometric mean of the normalized scores is computed as the final score for the tile. The resulting per-tile scores are then summed to create the final metric. A reference system is commonly used for normalization when computing benchmark scores.

5 EVALUATION

In this section, at first we evaluate the control server virtualization using different workloads. In the second step, we evaluate three access methods. In the third step, we evaluate the maximum level of consolidation in the prototype model and finally, we present v-motion functionality test (migration to different physical machine) and discuss the collected results.

5.1 Evaluation of Control Server

In this evaluation, we have compared CPU utilization percentage, memory usage, disk utilization and I/O rate in different workloads. Since the obtained graphs and results are extensive, we have solely focus on CPU utilization. We have considered two testing groups including the non-virtual (physical resource) and virtual model. In collection of non- virtual experiments, five different workloads programs are installed and the average rate of CPU utilization is measured for each test. On the other hand, in collection of virtual experiments, similar workloads are applied and similar parameters are measured. It should be noted that effect of hypervisor or virtualization software appears in the results and behave as an overhead for the physical system. However this effect is not a constant value for all scenarios and depends on the existing workload. In the following part, we evaluate the effect of five workloads separately.

1) *Database Workload*. To evaluate the effect of virtualization on CPU utilization in accessing to the data base workload, we have prepared the database scenario based on section 4 and the necessary software based on appendix A. Result of evaluation is given in Figure 2.

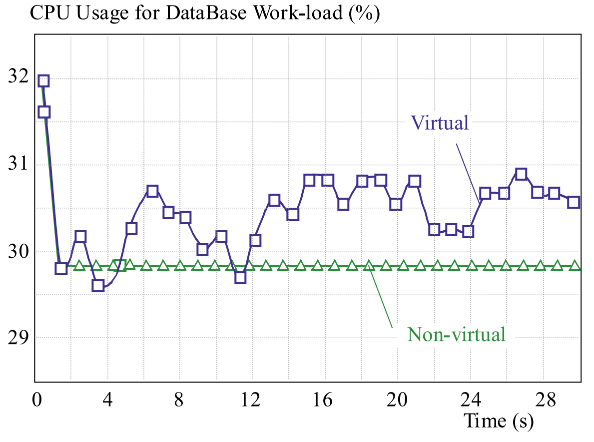

Fig. 2. Comparison of CPU utilization for database workload in virtual and non-virtual systems

As it is shown in Fig. 2, result of evaluation for database workload proof that in virtual model, CPU utilization is higher than non-virtual model. This advantage continues during the system execution.

2)*Mail Server Workload*. In this evaluation model, the mail server workload is applied to the both real and virtual systems and the results are shown in Fig. 3.

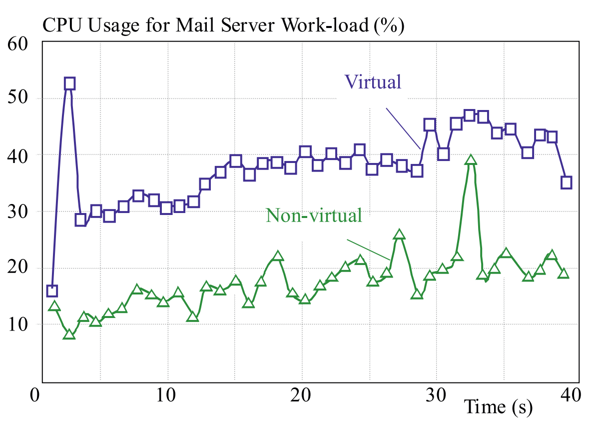

Fig. 3. Comparison of CPU utilization for mail server workload in virtual and non-virtual systems

Results show that for Mail server workloads, CPU utilization percentages in virtual systems are much higher than real systems. This advantage exists in different functions of an e-mail service.

3) *Java Server Workload*. In next evaluation scenario, Java workloads are applied to the system. We have evaluated Java workload to the virtual and non-virtual systems and the results are shown in Fig. 4.

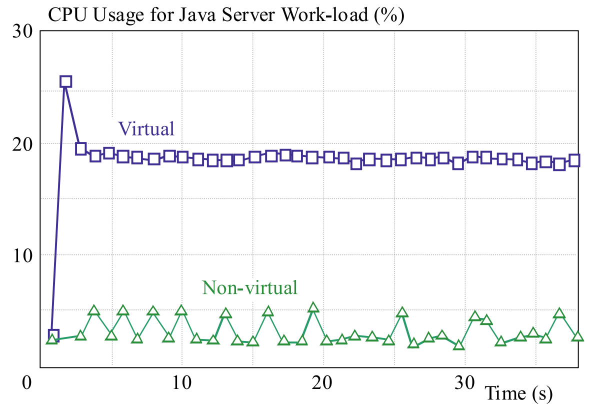

Fig. 4. Comparison of CPU utilization for Java server workload in virtual and non-virtual systems

As it is shown in Fig. 4, CPU utilization for virtual Java server workload is several times higher than non-virtual workload and this advantage is permanent during the system processing.

4) *Web Server Workload*. In this evolution, Web Server Workload is applied to the virtual and real test scenario. The results of evaluation and CPU usage percentages are given in Fig 5.

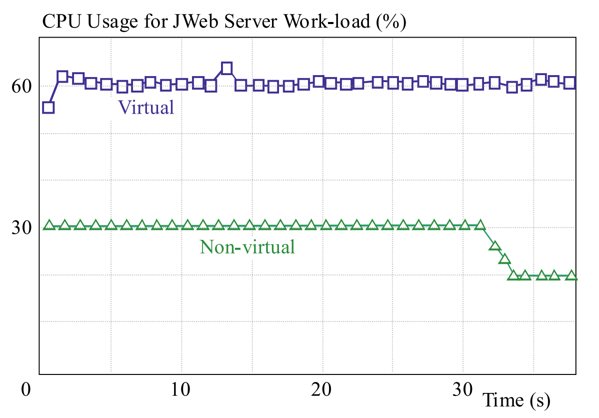

Fig. 5. Comparison of CPU utilization for Web server workload in virtual and non-virtual systems

Obtained results certify that CPU usage in virtual model is almost two times higher than in real one. It is a good advantage for using virtualization in web applications.

5) *Standby Server Workload*. In the last evaluation scenario, standby server workload is considered for evaluation of CPU utilization. This scenario is for comparing the effect of virtualization technique on CPU usage against the real system. Obtained results in Fig. 6 show that virtualization has positive effect on improvement of CPU utilization during the standby workload.

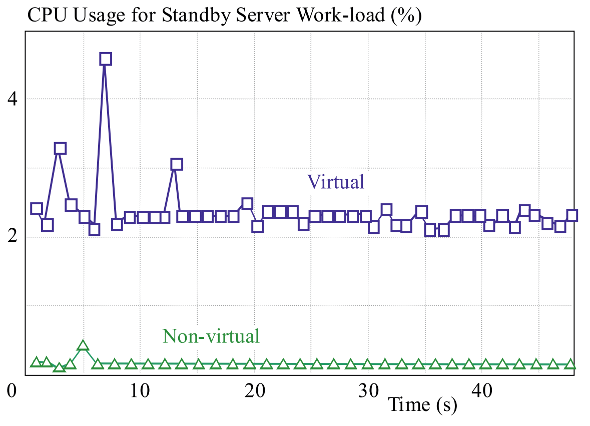

Fig. 6. Comparison of CPU utilization for Standby server workload in virtual and non-virtual systems

5.2 Access System

To evaluate the behaviour of three access methods and compare the advantage and disadvantage of each scenario in details, we have considered four traffic models using OLTP database, Data Streaming, File server, and Web server which are popular loads in data center applications.

1) *OLTP Database*. OLTP is a special data base with an online transaction processing. In this part, we have considered three parameters of IOps read/write, response time and CPU utilization in different data block size.

a) *IOps Read/Write*. In this evaluation, we have compared the IOps read and write for the OLTP database. The hardware model and software are as mentioned in Sections 3 and 4. The evaluations results are given in Fig. 7.

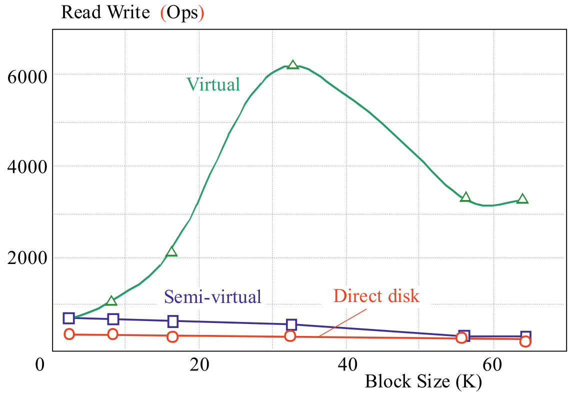

Fig. 7. Comparison of IOps Read/Write for OLTP Data base server in three scenarios

Results of IOps for OLTP data base show that the average of I/O operations (per second) for reading and writing in virtual storage has increased several times compared to the semi-virtual and direct methods. This parameter is almost constant in direct storage and semi-virtual regardless of block size. On the other hand, in virtual model the IOps read/write are sensivive to block size and in our prototype model between 25 and 40 k of block size, the IOps is maximum. IOps depends on many parameters which may be different in each system such as balance of read and writes operations, random access pattern, number of simultaneous processes, queuing length, queuing size, and so on. As a result, the read/write IOps level is increased sharply in virtualization model where the maximum value are limited in a specific block size and drop after certain size. This behavior depend on the test environment and technical specification of the system. Obtain results confirm that the IOps read/write in virtualization model is significantly higher than other two methods.

b) *Average Response Time*. To compare the ART in different scenario, we have considered the average read and write response time for OLTP database server. The hardware and software models are the same as the previous test scenario. Results of ART for OLTP database in Fig. 8 shows that the Average Read/Write Response Time in virtual storage is reduced quickly. It confirms that the virtual storage system responses to the read and write instructions more quickly than the direct access and semi-virtual access methods.

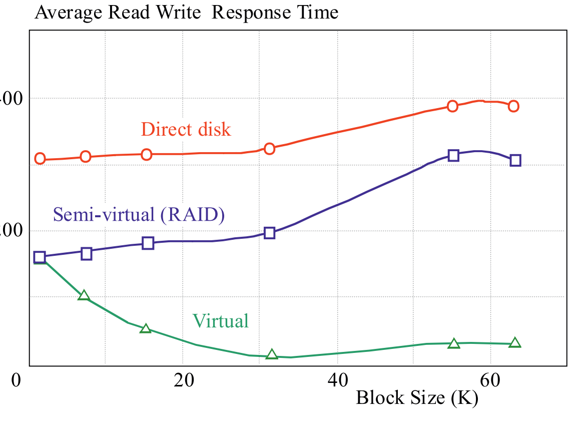

Fig. 8. Comparison of average Read/Write response time for OLTP database server

Average Read/Write Response Time depends on the data block size where in both direct storage and RAID models, the ART increases continuously in a limited level. On the other hand, in virtual storage, the average read/write response time is decreases to a minimum level and doesn't significantly change by block size variation. As a result, the average I/O response time is reduced significantly by using virtual technology. In addition the variation of average response time is not so sensitive to block size in virtual model where in the two other methods, the ART is more sensitive in larger block sizes.

c) *CPU Utilization*. The third evaluation factor which is very critical for database access is CPU utilization percentage. Results of evaluation for the OLTP data base show that the CPU utilization has increased in virtualization model compares to the other two methods as it is shown in Fig. 9. Also in this method, the utilization has increased sharply until a maximum level which depends on the technical specification of the control system. On the other hand, increasing the data block size increases the CPU load and limits the utilization factor. This limitation depends on the available resources and specification of traffic, where the larger block size data increases the load and limit the CPU utilization percentage. On the contrary, the CPU utilization rate has small variation by changing the block size in direct and RAID models. As a result, using a virtual technique will increase the CPU utilization factor, where it is limited by the block size. Using the maximum CPU utilization increases the system sensitivity for the heavy workloads where the overload resulted from large block size, reduces the utilization factor.

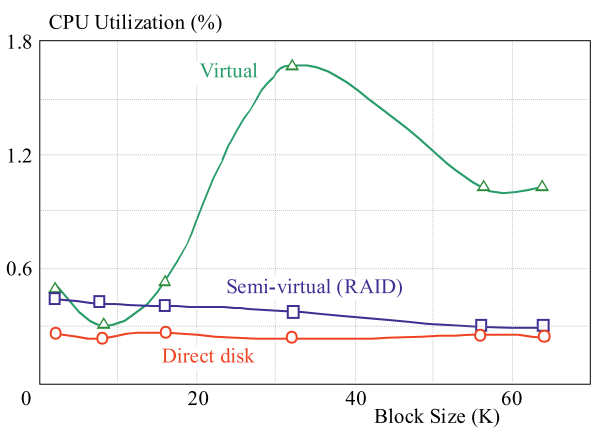

Fig. 9. Comparison of CPU utilization percentage for OLTP database server in the three scenarios

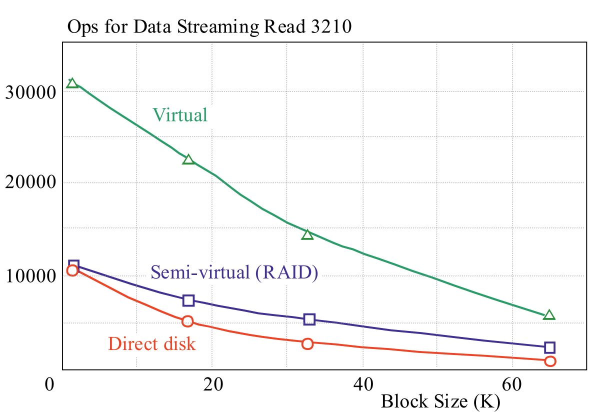

Fig. 10. Comparison of IOps Read for Data streaming

2) *Data Streaming Evaluation*. In this part we have generated data streaming traffic and applied to the storage system. We have considered IOps read, IOps write and average transition time as the three evaluation parameters while we have changed the data block size in each step. This arrangement used for three scenarios of direct, RAID and virtual models.

a) *IOps Read*. Data stream is a sequence of digitally encoded coherent data used to transmit or receive information that is in transmission. It is a useful traffic model for evaluating a storage system. Figure 10 shows IOps (or average number of reading action per second) results of data streaming read for three scenarios of virtual, semi-virtual and direct storage.

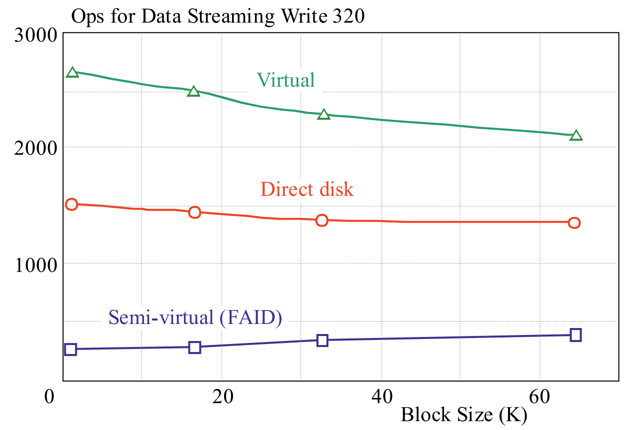

Fig. 11. Comparison of IOps Write for Data streaming

Results show that in general case, the IOps rate decreases while increasing the block size in three scenarios. Also the initial value of IOps rate in virtual model is more than rate in RAID and direct storage model. In addition, in direct access and RAID, the Read IOps is almost the same where in the virtual model it has noticeable improvement. As a result, the virtual storage method increases the reading IOps rate more effectively compare to the other two methods.

b) *IOps Write*. Figure 11 shows the write IOps results of data streaming for the three scenarios. Results show that the IOps write rate directly depends on access method.

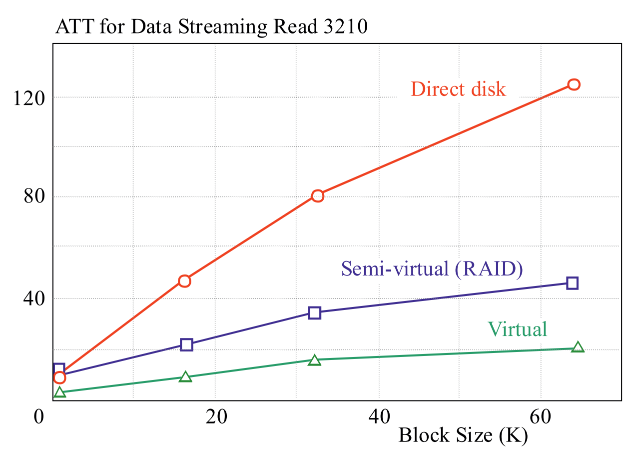

Fig. 12. Comparison of average transaction time for Data Streaming

Table 3. File server application for direct, semi-virtual and virtual with RAID-5 and 32 I/O threads

Scenarios	R/W IOps	R/W Mbps	ATT/ ART	%CPU Utilization
Direct Storage	405.6	4.4	315.4	0.23
Semi-Virtual (RAID array)	843.8	9.1	88.5	0.52
Virtual Storage	6967	75.3	19.8	1.9

Table 4. Web server application for direct, semi-virtual and virtual with RAID-5 and 32 I/O threads

Scenarios	R/W IOps	R/W Mbps	ATT/ ART	%CPU Utilization
Direct Storage	412.4	6.3	310.2	0.27
Semi-Virtual (RAID array)	1334.1	20.3	95.9	0.89
Virtual Storage	7811.2	119.7	16.4	2.06

The IOps write shows different levels in three scenarios where in the non-virtual evaluation; variation is very low in different block size. On the other hand, in virtual system the write IOps will decrease more than other graphs. Results show that the write IOps rate is highly depends on the access method where the virtual method is on top. Moreover, virtual model is more sensitive to block size variation compare to other methods.

c) *Average Transaction Time*. In this part we have applied the Data Streaming traffic to the storage system and collect the results for measuring the average transaction time. Figure 12 shows ATT (Average executing time for each transaction) results of data streaming read for the three scenarios of virtual, semi-virtual and direct storage. Results show that average transaction time in virtual model is less than RAID and direct storage methods. The average transaction time has improved when the block size becomes larger. Therefore, storage virtualization increases the execution speed of transactions and has a direct effect on system performance and transaction time.

3) *File Server Evaluation*. In this part, we have compared the effect of file server transaction on access models. Similar traffic are applied to three scenarios and the average of IOps read/write, Mbps read/write, average response time, and CPU utilization are calculated. This evaluation had repeated for different block size and the total average are calculated for each scenario. Results are shown in Table 3.

Results of evaluation certifies that for Read/Write IOps, Read/write Mbps, CPU utilization and ATT parameters, the virtual storage scenario shows better outcome with noticeable improvement compare with other two methods.

4) *Web Server Evaluation*. In this evaluation, we have considered the Web Server application as the reference model for comparison. Web applications have been operational in three scenarios and the IOps read/write, Mbps read/write, average response time and, CPU utilization are considered as the evaluation parameters. The evaluations are repeated for different block size and 32 threads as mentioned in previous sections. Results of evaluation are shown in Table 4.

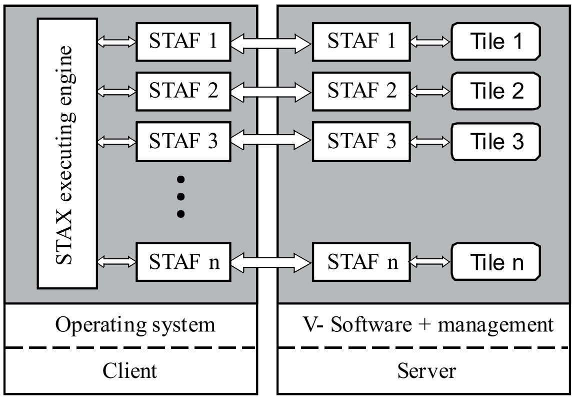

Fig. 13. Software model for test measurement in maximum consolidation

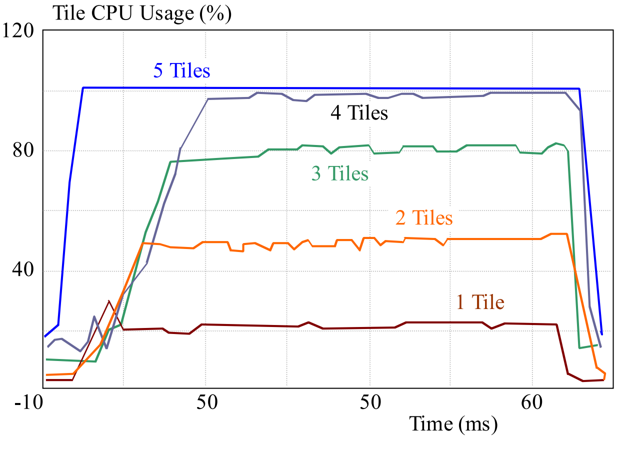

Fig. 14. CPU usage value

Obtained results for the above parameters show that virtual storage has better improvement and noticeable value compares to the direct and RAID storage systems.

5.3 Maximum Level of Consolidation

In the third part of evaluation, we have measured maximum level of consolidation. In other words, this experiment is determined the maximum number of tiles that can place on the virtualized system without significant damage in performance. Software model for testing the maximum level of consolidation is shown in Fig. 13. Implementation model is based on VMmark software architecture. In VMmark model, six virtual machines (workloads) are considered as a unit which is called tile unit. VMmark methodology uses the STAF and STAX tools.

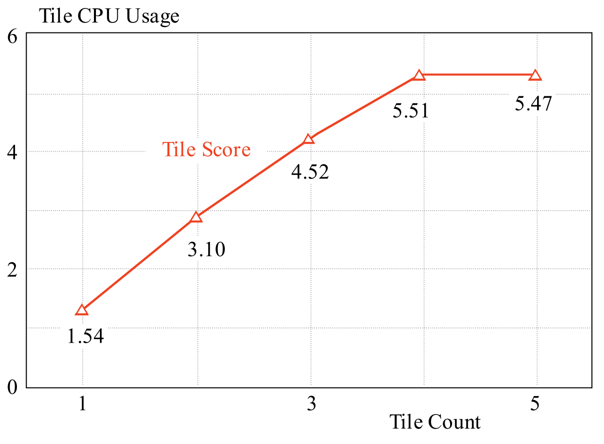

Fig. 15. Tile Score

STAF (Software Testing Automation Framework) is an automation framework which creates and manages an automated test cases and test environment. STAF externalizes its capabilities through services.

A service provides a set of functionality such as, logging, process invocation, etc. STAF Proc is the process that runs on a machine, call a STAF Client which accepts requests and routes them to the appropriate service. Requests come from STAF Server and STAF works in a peer environment where each machine can make requests of services on other machines. STAX (STAF Execution Engine) is an XML-based execution engine implemented as an external STAF service. STAX was designed to make it significantly easier to automate the workflow of your tests and test environments. STAX accepts the job definitions in form of XML documents. Fundamentally, these job definitions allow you to specify the processes and STAF commands necessary to perform the job. STAX provides a wealth of expressive functionality on top of this, making it easy to implement, manage, track, and monitor the jobs. STAX installed on the prime client as it is shown in Fig. 14. n STAF clients and n STAF Servers work in a peer to peer environment according to n Tiles where STAFs client make requests of services on other machines. In this experiment, we increased number of tiles until process power of the physical host machine reaches to the maximum value and the system denied accepting any new tile for this collection. In this stage, the maximum number of tiles and number of simultaneous virtual machines on physical host are considered as the maximum level of consolidation. From this threshold, adding any new number of tiles increase CPU usage until maximum level where exceeding the number of tiles does not increase the CPU utilization and decrease the tile score.

Therefore, we increase number of machines and the system will record the results of efficiency indicators. Changing the value of service efficiency indicators in each stage are suitable criteria for assessing performance of virtual machine monitor (VMM) on the sample hardware. To compute the score of tiles; first, set of one tile with six standard workloads run on the virtual server. We compute each workload's normalized scores by dividing the score of each workload by reference score of the workload. Then combine the normalized scores using a geometric mean, the score of a tile is for reporting, which includes the score as well as individual scores for the workloads (both raw and normalized). As the tile score computation contains a performance factor, it is fair to view tiles as a measure that represents both performance and scalability. CPU utilization percentage and tile score of Tiles are shown in Figs. 14 and 15.

Figure 14 shows CPU utilization percentage for five different tiles. It starts from minimum value for one tile and by adding new tiles; CPU percentage will increase until certain value that depends on system specification. In our test, maximum value for CPU percentage is five tiles.

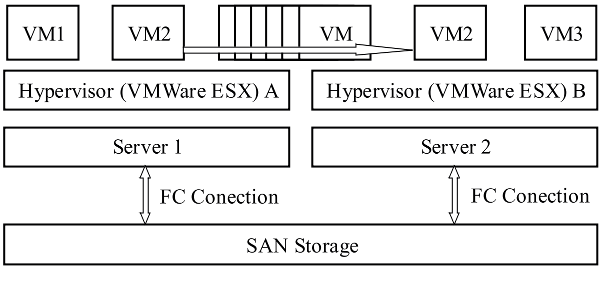

Fig. 16. Test software model to transfer the live virtual machine

Figure 15 shows tile score value for five different tiles. Evaluation is based on previous test and increases from one to five tiles. Maximum tile score is for four tiles and by adding the fifth tile, the tile score value has been reduced gradually. It indicates that in our experiment, maximum level of consolidation is four.

5.4 V-Motion Functionality Test

V-Motion means transfer an active virtual system on different physical machines. For this experiment, we have installed three sample virtual machines. The first virtual machine is installed on hardware A, and the third virtual machine on hardware B. The second virtual machine has installed in hardware A and should be able to move to hardware B. We have set parameters which can automatically manage different resources for this movement in the system. When system's workload exceeded from pre-defined values, one of virtual machines automatically move from physical machine A to physical machine B and the system parameters are measured. A test software model to transfer the live virtual machine is shown in Fig. 16.

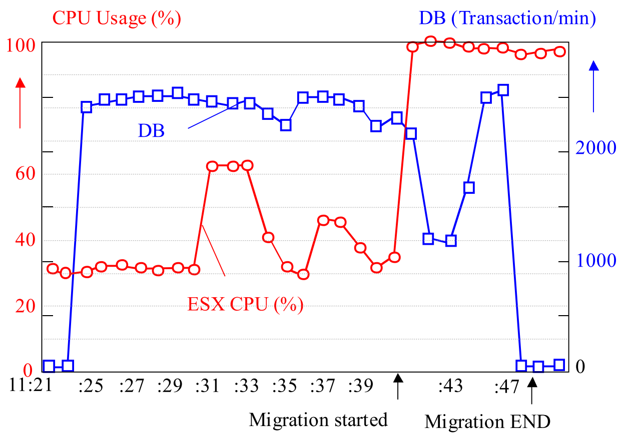

Fig. 17. Average CPU usage in live Migration on ESX1 (Aggressive Mode)

Results of test V-Motion is shown in Figs. 17 and 18. In this evaluation, a data base system has been selected for virtual server (VM2). Based on this model, VM2 is transferred from machine A (ESX1) to machine B (ESX2). Performance results of CPU usage percent for the DB, ESX 1 and virtual machine performance of DB during the transition is shown in Fig. 17. Also, performance results of CPU usage percent for the DB, ESX 2 and virtual machine performance of DB during the transition is shown in Fig. 18.

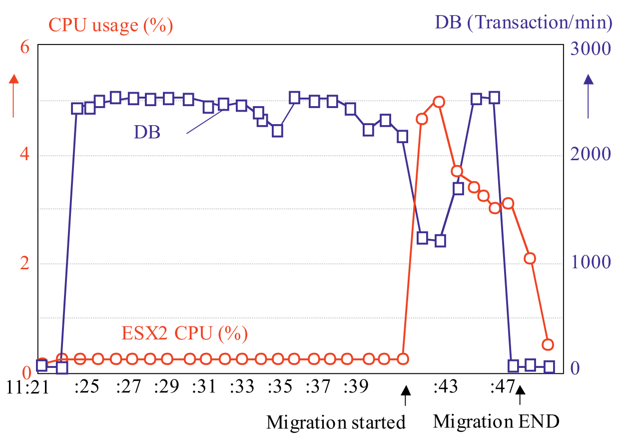

Fig. 18. Average CPU usage in live Migration on ESX2 (Aggressive Mode)

As it is shown in Fig. 17, CPU usage of ESX1 and DB Tr/Min rate are given before and after of moving DB from ESX 1 to ESX 2. The ESX 1 CPU percentage is working in a normal condition and from specific point increase to an unacceptable level, in this case the movement has started. At the same time, the DB Tr/Min will reduce in a moving interval and transfers the load to CPU 2. As a result, moving an active DB from one machine to new machine will reduce the DB transaction rate in the moving interval and quickly back to the normal transfer rate. On the other hand, based on Figure 18, the CPU usage in ESX 2 is in a normal operation, where the movement has started. In this case, the CPU usage in ESX2 will increase during the movement interval. At the same time, the DB (TR/Min) is working in a normal operation where in the process of V-Motion, the DB (Tr/Min) will reduce in a short interval and will move to the new machine. In this case, the CPU usage of ESX2 will increase to support the DB transfer jobs.

6 CONCLUSIONS

Virtualization technique is becoming more and more popular in datacenter applications. This paper presents an evaluation scenario for virtual server and access model. Results of evaluation with considering different criteria confirm that server virtualization techniques have higher throughput and CPU usage as well as better performance with noticeable agility. Also server virtualization can significantly reduce resource consumption while improve the system performance. On the other hand, we have focused primarily on the performance of available technologies (*ie* virtual, semi- virtual and non-virtual) for accessing to the storage systems. We have selected the IOmeter software for benchmarking and characterized different workloads in three dimensions of read I/O percentage, random I/O percentage and block size. We specified four models of workloads including the database, data streaming, file server, and web server which are more popular in data center applications. Results show that virtual storage systems perform better IO transfer rate and I/O operation speed in different applications. Also, the virtual storage system optimizes the CPU utilization, improves the average transaction/response rate, and increases the IOps/Mbps read/write in different applications. On the other hand, capability for moving a virtual machine in different hardware systems increases the system reliability and improve business continuity plan. The obtained results certify that new strategy and investment for storage systems should concentrate on full virtual techniques rather than other non-virtual systems.

Acknowledgement

The author acknowledges the assistance of Mr. Davood Maleki.

References

[1] AHMADI, M. R.—MALEKI, D.: Effect of Virtual Techniques in Data Storage Access, The 24th IEEE International Conference on Advanced Information Networking and Applications, Australia, 2010, pp. 91–96.

[2] WALTERS, J. P.—CHAUDHARY, V.—CHA, M.—GUERCIO, S.—GALLO, S.: , 22nd International Conference on Advanced Information Networking and Applications, IEEE (2008), 861–868.

[3] SMITH, J. E.—NAIR, R.: The Architecture of Virtual Machines, Computer **38** No. 5 (2005), 32–38.

[4] GOLDWORM, B.—SKAMAROCK, A.: Blade Servers and Virtualization, Transforming Enterprise Computing while Cutting Costs, Wiley Publishing, Inc, 2007.

[5] ZHANG, G.—SHU, J.—ZHENG, W.: Design and Implementation of an Out-of-Band Virtualization System for Large SANs, IEEE Transaction on Computer **56** No. 12 (Dec 2007).

[6] CHAMBLISS, D.—ALVAREZ, G.: Performance Virtualization for Largescale Storage Systems, Proceedings of the 22nd International Symposium on Reliable Distributed Systems (SRDS'03), 2003.

[7] CLARK, T.: Storage Virtualization: Technologies for Simplifying Data Storage and Management, Addison-Wesley Professional, Mar 2005.

[8] AHMAD, I.—ANDERSON, J. M.: An Analysis of Disk Performance in VMware ESX Server Virtual Machines, VMware Inc., Palo Alto CA 94304, IEEE 2003.

[9] LUMB, C.—MERCHANT, A.—ALVAREZ, G.: Virtual Storage Devices with Performance Guarantees, Proceedings of the Conference on File and Storage Technologies (FAST), 2003, pp. 131–144.

[10] KEETON, K.—VEITCH, A.—OBAL, D.—WILKES, J.: I/O Characterization of Commercial Workloads, Proc. Third Workshop on Computer Architecture Evaluation Using Commercial Workloads (CAECW-00), 2000.

[11] GOMES, M. E.—SANTONJA, V.: Self-Similarity in I/O Workloads: Analysis and modeling, Proc. First Workshop on Workload Characterizations, 1998, pp. 99–104.

[12] Open Source Development Lab. IOmeter I/O performance analysis tool, 2003.

[13] Intel Crop, "IOmeter," http://www.IOmeter.org, 2004.

[14] TEIGLAND, D.—MAUELSHAGEN, H.: Volume Managers in Linux, Proc. Usenix Ann. Technical Conf., 2004.

[15] LEE, D.—CHO, S.—PARK, J.—PARK, P.: Implementation of Hardware RAID and LVM-Based Large Volume Storage on Global Data Center System of International GNSS Service, Korea Astronomy and Space Science Institute, Daejeon, Korea, 2005.

[16] JANG, J.—HAN, S.—KIM, J.—PARK, S.: A Performance Evaluation Methodology in Virtual Environments, The 7th IEEE International Conference on Computer and Information, Fukushima, Japan, Oct 2007, pp. 351–356.

[17] UMENOI, H.—PARAYNOT, M.: Performance Evaluation on Server Consolidation Using Virtual Machines, SICE-ICASE International Joint Conference, Bexco, Busan, Korea, 2006 Oct, 18-21, pp. 2730–2734.

[18] UMENO, H.—PARAYNO, M. L. C.—TERAMOTO, K.—KAWANO, M.: Performance Evaluation on Server Consolidation Using Virtual Machines, SICE-ICASE International Joint Conference, Bexco Busan, Korea, 2006 Oct, 18–21.

[19] AHMAD, I.—ANDERSON, J.—HOLLER, A. M.—KAMBO, R.: An Analysis of Disk Performance in VMware ESX Server Virtual Machines, in Workload Characterization, 2003. WWC-6. 2003 IEEE International Workshop on, Oct 2003, pp. 65–76.

[20] WALTERS, J. P.—CHAUDHARY, V.—CHA, M.—GUERCIO, S.—GALLO, S.: A Comparison of Virtualization Technologies for HPC, 22nd Internat. Conf. on Advanced Information Networking and Applications, IEEE, 2008, pp. 861–868.

[21] VMware VMmark Benchmarking Guid, VMmark 1.1, VMware Inc, 2008, http://www.vmware.com/support/.

[22] STEINDER, M.—WHALLY, I.—CARRERA, D.—CHESS, D.: Server Virtualization in Automatic Management of Heterogeneous Workloads, IEEE.

[23] XIANGHUAU, X.—ZHOU, F.—YUCHENG, J. W.: Quantifying Performance Properties of Virtual Machine, 2008 International Symposium on Information Science and Engineering, IEEE, 2008.

Received 24 January 2011

Mohammad Reza Ahmadi received the BSc and MSc degrees in Electrical Engineering and Communication Systems from KNT University of Technology in 1986 and 1990 respectively. He received his PhD degree in Communication Networks from Tokyo Institute of Technology, Tokyo, in 1997. Currently he is the project manager and researcher in IT department of Research Institute of ITC, in CSRI. His research interests are resource optimization in data centers, virtualization and cloud computing techniques, and network security focus on detection systems.

APPENDIX A

#	Software Name	Description
1	VMware ESX Server 3.5	Creator Virtual Machine
2	Microsoft Windows 2003 Standard Edition 32bit	Operating System in Clients
3	VMmark Harness V1.1	Benchmark software
4	SPECweb2005 client	Software Application and workload manufacturer for web server [13]
5	SPECjbb2005 monitor	Software Application and workload manufacturer for Java server [14]
6	IBM's STAF kit v3.3.3 STAF Event Service	testing automatic framework (Environment of test command writing)
7	STAX Execution Engine v3.3.6	Executive machine for STAF
8	Microsoft Exchange 2003	software of workload producing for mail server
9	Microsoft Outlook 2003	Software of email server application
10	Microsoft Loadsim 2003 v6.5.7775.0	software of workload producing for data base server
11	Oracle BEA JRockit 1.5.0_17 JVM JDK	Software of data base server application
12	Dbench v1 (Modified by VMware for use the VMmark benchmark	software of workload producing for file server