Utility Analysis for Internet-Oriented Server Consolidation in VM-Based Data Centers

Ying Song, Yanwei Zhang, Yuzhong Sun
Key Laboratory of Computer System and Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
e-mail: {songying, zhangyanwei}@ncic.ac.cn; yuzhongsun@ict.ac.cn

Weisong Shi
Wayne State University
Detroit, USA
e-mail: weisong@cs.wayne.edu

***Abstract*—Server consolidation based on virtualization technology will simplify system administration, reduce the cost of power and physical infrastructure, and improve utilization in today's Internet-service-oriented enterprise data centers. How much power and how many servers for the underlying physical infrastructure are saved via server consolidation in VM-based data centers is of great interest to administrators and designers of those data centers. Various workload consolidations differ in saving power and physical servers for the infrastructure. The impacts caused by virtualization to those concurrent services are fluctuating considerably which may have a great effect on server consolidation. This paper proposes a utility analytic model for Internet-oriented server consolidation in VM-based data centers, modelling the interaction between server arrival requests with several QoS requirements, and capability flowing amongst concurrent services, based on the queuing theory. According to features of those services' workloads, this model can provide the upper bound of consolidated physical servers needed to guarantee QoS with the same loss probability of requests as in dedicated servers. At the same time, it can also evaluate the server consolidation in terms of power and utility of physical servers. Finally, we verify the model via a case study comprised of one e-book database service and one e-commerce Web service, simulated respectively by TPC-W and SPECweb2005 benchmarks. Our experiments show that the model is simple but accurate enough. The VM-based server consolidation saves up to 50% physical infrastructure, up to 53% power, and improves 1.7 times in CPU resource utilization, without any degradation of concurrent services' performance, running on Rainbow - our virtual computing platform.**

Keywords-utility analysis; server consolidation; model

I. INTRODUCTION

Virtualization offers opportunities not only to better isolation and manageability but also to on-demand resource provision for server consolidation. There are many efforts focusing on virtualization, such as resource virtualization [37][38], dynamic deployment of virtual machine [39][40], on-demand resource allocation among the hosted virtual machines (VMs) [5][20][21][23], and so on. These works lead to improvements in the performance of virtualization and resource utilizations. Many researchers have argued that virtualization technology, such as virtual machine, will be ubiquitously used in cloud computing for server consolidation. However, the trend of using virtualization for server consolidation by enterprise data centers is not as popular as it has been expected. We attribute this phenomenon to the performance unpredictability [24], including unpredictability of requests arrival distribution, service performance, etc.

Figure 1 (a) and (b) illustrate the scenarios of multiple services hosting on dedicated servers and on consolidated servers, respectively. The former is the familiar manner of service deployment in today's data centers. In such case, one server could not be shared by more than one service, even though it is idle. The key advantage of using dedicated servers is no interactions between services. However, the waste of resources and power is its obvious disadvantage. Server consolidation can improve resource utilization and save power. At the same time, encapsulating the concurrent services into various VMs can isolate those services from interacting on each other. The fluctuations and differences on resource requirements of the concurrent services offer opportunities to server consolidation in the improvements of resource utilization and saving power. Figure 2 illustrates the results of

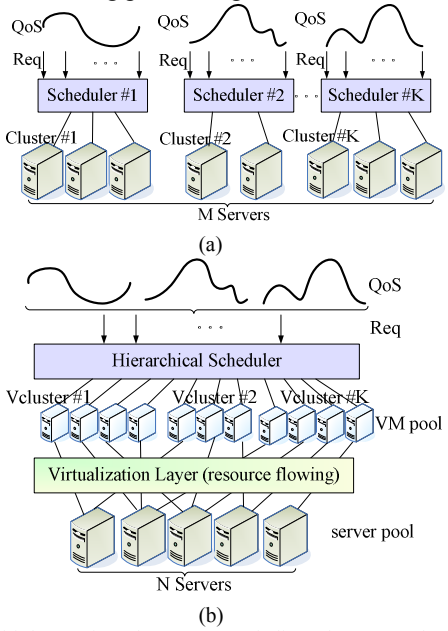


Fig. 1. Multiple services host on (a) dedicated servers (b) consolidated servers.

consolidating three applications with various features to the shared servers. The peak of consolidated workloads will not higher than the sum of the dedicated workloads' peaks. Thus, consolidated workloads may need fewer physical servers than dedicated workloads do with the same loss probability of requests. How many servers are needed to guarantee performance of the consolidated workloads in some probability level (the line in Figure 2 (b))? Various consolidations of workloads differ in saving power and physical servers for the infrastructure. The impacts caused by virtualization to those concurrent services are fluctuating considerably which may have a great effect on server consolidation. Thus, the administrators and designers of Internet-oriented data centers do not definitely realize their potential revenue using virtualization for server consolidation instead of using dedicated servers to host their services. To our knowledge, no other efforts show how much power and how many physical servers are saved for the Internet-oriented data centers using virtualization for server consolidation. This is a very interesting problem, and this paper addresses it based on an analytic model.

There are a few research efforts [20][21][12][36] having addressed utility management to save power and improve quality of services (QoS) using the techniques of on-demand resource management and dynamically turning on/off servers for server consolidation in VM-based data centers. However, all these efforts are reactive, making decisions during the process of running the services. Such dynamic controls of resource allocation and VM mapping are not enough to guide the widely use of virtualization. Furthermore, no other work focuses on planning the scale of an Internet-oriented data center when multiple services are to be consolidated into a VM-based sharing platform before running these services. Our work addresses this challenge, complementing very well to these previous efforts. The combination of the former reactive works and this work guides the plan and management of VM-based data centers, which undoubtedly contributes a lot to the wide use of virtualization in data centers.

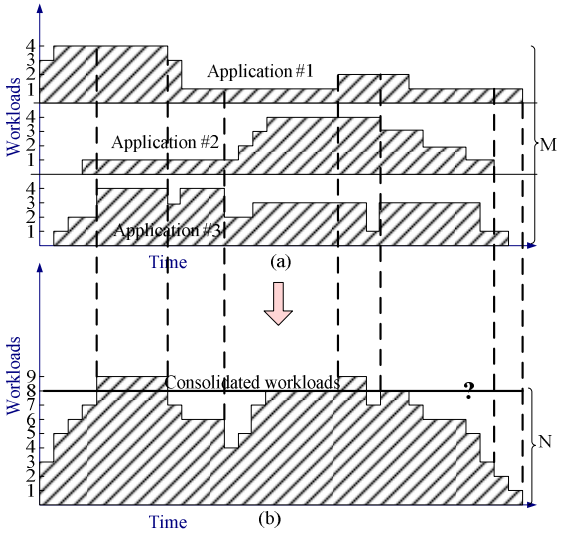


Fig. 2. Workloads offered to (a) the dedicated servers are consolidated to (b) the consolidated servers.

We revisit the relationship between services and resources, taking the impact of virtualization into account. In order to address the above challenge, this paper proposes a utility analytic model for Internet-oriented server consolidation in VM-based data centers, modelling the interaction between server arrival requests with several QoS requirements, and capability flowing amongst concurrent services based on the queuing theory. According to features of those services' workloads, this model can provide the upper bound of consolidated physical servers needed to guarantee QoS with the same loss probability of requests as in dedicated servers. At the same time, it can also evaluate the server consolidation in terms of power and utilization of physical servers, the overhead of virtualization, and the algorithms of on-demand resource allocation in the VM-based data centers.

We implemented a Xen-based prototype called Rainbow [22][23] used to verify our utility analytic model via a case study comprised of one e-book database service and one e-commerce Web service simulated respectively by TPC-W and SPECweb2005 benchmarks reflecting resource demands of services in a real enterprise environment. First we analyse the performance impact of virtualization on CPU and I/O using these services. Then, we do a case study to verify our model. Our experiments show that the model is simple but accurate enough. In our case study, server consolidation consumes up to 50% less physical infrastructure, up to 53% less energy, and 1.7 times higher CPU utilization than traditional dedicated servers, without any degradation of QoS.

This paper has the following three contributions: (1) We propose a utility analytic model for server consolidation to predict the scale of VM-based data centers; (2) We verify the model by evaluating the impact of virtualization in the context of two real workloads; and (3) We evaluate the power consumption for VM-based server consolidation.

The rest of this paper is organized as follows. Section II discusses related work. Section III introduces utility analytic model. Section IV discusses the experimental results. We conclude in section V.

II. Related Work

Nowadays, virtualization and power are the two hot research fields in data centers. Server consolidation combines these two fields. With virtualization rapidly regaining popularity in the past few years, there have been a number of recent papers on the server consolidation using virtualization, such as the analysis on performance overheads of virtualization [2][3][4], the performance prediction of virtual environments [7][8][9], and the on-demand resource allocation [5][6][19][20][21][22][23] for VM-based data centers, etc. Server consolidation also contributes to reduce power cost [12][13][36], although hardware technologies are popular in data centers [32][33][34][35], and also some power budget technologies[43].

A. Virtualization for Server Consolidation

Performance evaluation: Reference [2] evaluated two representative virtualization technologies, Xen and OpenVZ,

in the context of server consolidation for a multi-tiered service. They considered the application-level integral performance impacted compared to its performance on a base Linux system for such multi-tiered service. However, different tiers of a multi-tiered service have various characteristics on resource requirement, which results in various performance impacts compared to its performance on a base Linux system. Our work evaluates the negative impact of virtualization on various services or various tiers of the multi-tiered service, separately. Other studies also provided performance evaluation of applications running on Xen [3][4]. In common, all these efforts were based experimental evaluation of Xen and its application. Reference [7] proposed simple queuing models for predicting the performance that applications, currently running on Linux system, would achieve if migrated to a Xen virtual system, with same hardware configuration. They only focused their research on the Web server, not the server consolidation, and only used the CPU busy time to evaluate the application performance. More recently, queuing network models for performance prediction of virtual environments were proposed in [8][9]. These works can help to determine the impact of an application running on a virtual machine, which is needed by our model.

Dynamic resource allocation: Reference [5] developed a communication-aware CPU scheduling algorithm to dynamically allocate CPU resources on the scenario of server consolidation. Reference [42] proposed the Entropy resource manager for homogeneous clusters, which performed dynamic consolidation based on constraint programming and took migration overhead into account. Reference [20][21][19] and our prior works [22][23] focused on the on-demand resource allocation for server consolidation in such VM-based data centers to improve QoS and resource utilization. All these efforts were based solely on experimental verification lacking of theoretical evaluation. Such theoretical evaluation is provided by our utility analytic model.

B. Energy Management in Data Centers

A large variety of power-saving proposals have been presented based on hardware technologies in the literature [32][33][34][35]. However, some authors [12][13][36] have argued that workload consolidation and powering off spare servers are effective ways to save power and cooling energy. Our work belongs to the latter.

Hardware technologies for saving power: A few research efforts [32][33] tackled the high energy consumed by server CPUs. Their approach was to conserve energy by using either dynamic voltage scaling or request batching under light load. Other efforts [34][35] addressed the energy consumption in the storage subsystem.

Energy management strategies: Many efforts [6][12][13][36][14][28][29][30][31] have examined energy management strategies in server clusters. These efforts tackled the high "base" power of traditional server hardware (i.e. the power consumption when the system is powered on but idle), by dynamically reconfiguring (or shrinking) the cluster to operate with fewer nodes under light load. For example, reference [6] provided an optimal dynamic plan of VM to physical server mapping over time with VM migration. All these works were reactive, which made decisions during the process of running the services. Only such dynamic control of turning on/off physical servers is not enough to guide the management of data centers for the administrators and designers. Our work helps to plan the scale of data centers before running any services, complementing very well to these previous efforts. The combination of the former reactive works and this work contributes to the wide use of VM-based server consolidation in data centers.

III. The Utility Analytic Model

Based on the Erlang's loss formula [41] in queuing theory, we design a utility analytic model to evaluate the upper bound of physical servers needed to guarantee the quality of the concurrent Internet services in the scenario of server consolidation for VM-based data centers. It models the interaction between arrival requests with several types of QoS metrics, and capability flowing amongst concurrent services. This model simply needs the average arrival rate of each service and the average serving rate of each resource in a physical server, without running these services. The outputs to this model are the relationships between the dedicated servers and the consolidated servers in the aspect of the number of servers (M and N represent the number of dedicated servers and the number of consolidated servers, respectively), the resource utilization of servers (U_M and U_N represent the utilization of dedicated servers and the utilization of consolidated servers, respectively), and the power consumption of servers (P_M and P_N represent the power consumption of the dedicated servers and of the consolidated servers, respectively), with the same loss probability of requests.

A. The Introduction to Erlang's Loss Formula

The service quality of a system can be evaluated by the loss probability of requests which has two methods to measure:

- The loss probability calculated by time p_n, which denotes the probability of no available servers within unit time;
- The loss probability calculated by requests B, which denotes the ratio of the number of loss requests to the number of arrival requests within unit time.

Erlang's loss formula:

$$B=p_n=E_n(\rho) \qquad (1)$$

where ρ denotes the traffic: $\rho=\frac{\lambda}{\mu}$, $p_n=\frac{\rho^n/n!}{\sum_{k=0}^{n}\rho^k/k!}$ and $B=\frac{\lambda p_n}{\lambda}=p_n=E_n(\rho)$.

According to ρ and B, we can calculate the upper bound of servers (n) using Erlang's loss formula with the following iterative method.

$$E_n(\rho)=\frac{\rho E_{n-1}(\rho)}{n+\rho E_{n-1}(\rho)}, \quad E_0(\rho)=1 \qquad (2)$$

When $E_n(\rho)\le B$ is satisfied firstly, n is the result.

B. The Utility Analytic Model

A utility analytic model is designed to evaluate the upper bound of physical servers needed to guarantee QoS of concurrent Internet services with the same loss probability of requests as in dedicated servers. At the same time, this model can also evaluate the server consolidation in terms of power and utilization of physical servers in the scenario of VM-based data centers, which is illustrated in Figure 3. Figure 3 (a) gives the arrival, waiting, and finish of requests for various services (using different colors to denote requests for various services) on the dedicated servers. In such case, requests for some service can not wait in queues of other services, and can not utilize the server resources exclusively used by any other services. However, in the consolidated servers (illustrated in Figure 3 (b)), any request waiting in its queue is despatched to a VM hosting the service accessed by it. All VMs serving a single service map to all physical servers (such mapping is showed by dashed lines located between the VM and the physical server). Thus, any request may be served at any physical server using any resource via resource flowing among VMs on the consolidated servers [5][21][23]. The requests despatched to VMs that map to the same physical server wait in integrated queue for some physical resource.

First, we introduce the assumptions we used to design the utility analytic model. Then, based on the Erlang's loss formula we express the problem of queuing in the two scenarios of dedicated servers and consolidated servers. Finally, we deduce the relationship between the dedicated servers and the consolidated servers in the aspect of the number of servers, the utilization of servers, and the power consumption of servers.

1) Assumptions: Our utility analytic model bases on the following four assumptions:

- *Homogeneous physical servers.* This assumption is used to simplify the expression of our model. Real-life data centers are almost invariably heterogeneous in terms of the performance, capacity, and power consumption of their hardware components. However, all the heterogeneous servers can be normalized to the homogeneous servers. For example, CPU of a server which has two 2.0GHz Quad-Core processors can be normalized to 1, then CPU of a server which has one 2.0GHz Quad-Core processor can be normalized to 0.5.
- *The arrival rate of each service follows a Poisson distribution.* Many researchers have discovered that user-initiated TCP sessions arrive at a WAN according as a Poisson process [10][11]. The only differences among these services are the average arrival rate (λ). We use λ_i to denote the average arrival rate of service i. The average serving rate of resource j for service i is denoted as μ_{ij}. The serving rate of each resource follows a general steady distribution, regardless of resources' damage.
- *Different types of resources, such as CPU and memory, do not interact on each other.* Although, many resources may have some interactions between one another in reality. For example, less memory may lead to more operations of disk I/O and CPU. In order to simplify our model, we ignore such interactions.
- *Servers provide service to requests on demand.* In queuing theory, one of the basic rules is whenever there is a request to be served, there are no servers being idle. Actually, in the VM-based server consolidation platform, the scheme of on-demand and fine-grained resource flowing amongst VMs just matches this rule. However, this assumption describes the optimal effects of on-demand resource allocation algorithms in VM-based data centers, ignoring the overhead of such resource reallocations. Thus, the utility analytic model can evaluate on-demand resource allocation algorithms via the close degree of QoS provided by the algorithms and provided by this model.

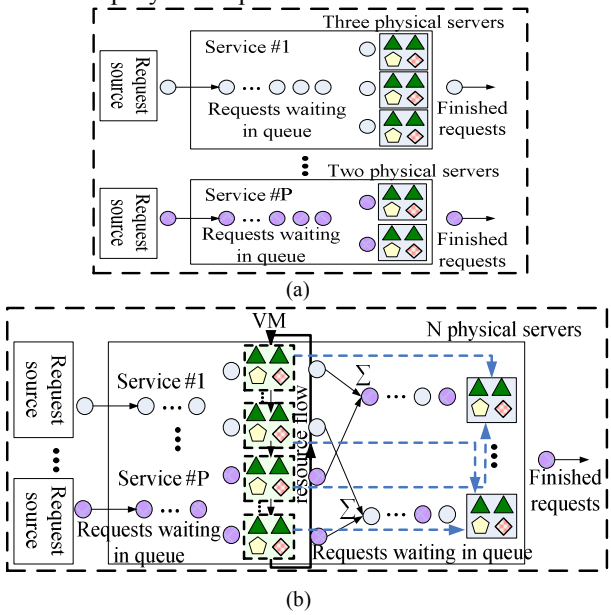


Fig. 3. The arrival, waiting, and finish of requests (a) in the scenario of dedicated servers and (b) in the scenario of VM-based server consolidation.

2) Problem Statement: We take P services and R types of resources into account in our model. We denote the number of dedicated servers to be M, and the number of consolidated servers to be N.

- In the scenario of dedicated servers for service i and resource j:

$$\rho_{ij} = \frac{\lambda_i}{\mu_{ij}}, \quad i=1,\ldots,P; j=1,\ldots,R \qquad (3)$$

- In the scenario of server consolidation for resource j and all the P services, we consider the average arrival rate for the consolidated workloads and the average serving rate (μ_j) of resource j for the consolidated workloads. The average arrival rate for the consolidated workloads is the summery of the average arrival rate of all services, namely, $\lambda = \sum_{i=1}^{P} \lambda_i$. The consolidated workloads also follow a Poisson process. We determine the average serving rate of resources j for the consolidated workloads according to the arrival rates of all the concurrent workloads, the serving rate of resource j for

each dedicated workloads, and the impact factor (***impact factor*** (*a*) denotes the ratio of the QoS provided by VMs to that provided by the native Linux) of virtualization on such resource. Each resource will serve the consolidated workloads after server consolidation. The probability of a request accessing service i is $\lambda_i / \sum_{j=1}^{P} \lambda_j$, while the serving rate of resource j for such request is $\mu_{ij}*a_{ij}$, taking the impact factor of virtualization (a_{ij}) into account. Thus, the average serving rate for the consolidated workloads can be expressed as follows.

$$\mu_j = \sum_{i=1}^{P} \frac{\lambda_i}{\sum_{i=1}^{P} \lambda_i} \times (\mu_{ij} \times a_{ij}) = \frac{\sum_{i=1}^{P} \lambda_i \mu_{ij} a_{ij}}{\sum_{i=1}^{P} \lambda_i} \quad (4)$$

where a_{ij} ($0<a_{ij}\leq 1$) denotes the impact factor of virtualization on resource j to service i, which reflects the degree of performance degradation of service i resulted from virtualization on resource j.

$$\rho_j^{'} = \frac{\lambda}{\mu_j} = \frac{\sum_{i=1}^{P} \lambda_i}{\frac{\sum_{i=1}^{P} \lambda_i \mu_{ij} a_{ij}}{\sum_{i=1}^{P} \lambda_i}} = \frac{(\sum_{i=1}^{P} \lambda_i)^2}{\sum_{i=1}^{P} \lambda_i \mu_{ij} a_{ij}} \quad (5)$$

3) Dedicated Servers vs. Consolidated Servers: Now we are in a position to compare the dedicated servers and consolidated servers from three perspectives, i.e., the number of servers, the utilization of servers, and the power consumption of servers.

(1) The number of servers

The above assumptions satisfy the conditions of Erlang's loss formula in queuing theory. Thus we use the Erlang's loss formula to solve the aforementioned problem.

Using the above iterative method to the Erlang's loss formula, we can give the process of calculating the number of servers (M and N, illustrated in Figure 4). Given the loss probability of requests (B), in the case of dedicated servers, we get n_{1j} for service #1 on resource j when $E_{n1j}(\rho_{1j})$ first satisfies $E_{nj1}(\rho_{1j})\leq B$; we get n_{2j} for service #2 on resource j when $E_{n2j}(\rho_{2j})$ first satisfies $E_{nj2}(\rho_{2j})\leq B$; … ; we get n_{Pj} for service #P on resource j when $E_{nPj}(\rho_{Pj})$ first satisfies $E_{nPj}(\rho_{Pj})\leq B$. In the case of server consolidation, we get N_j for all the services on resource j when $E_{Nj}(\rho_j^{'})$ first satisfies $E_{Nj}(\rho_j^{'})\leq B$. Each calculated number of servers is just for one type of resources. In order to assuring the performance of each service, the maximum number of servers for all types of resources should be the total required number of servers. Then we get the number of dedicated servers and the number of the consolidated servers as follows:

$$M = \sum_{i=1}^{P} \max\{n_{ij} \mid \forall j \in [1,...,R]\} \quad (6)$$

$$N = \max\{N_j \mid \forall j \in [1,...,R]\} \quad (7)$$

Input: B, λ_i, μ_{ij}, a_{ij}, $1\leq i\leq P$, $1\leq j\leq R$;
Output: M, N;
Algorithm: Begin
$\rho_{ij}=\lambda_i/\mu_{ij}$; $\rho_j^{'}=\frac{(\sum_{i=1}^{P}\lambda_i)^2}{\sum_{i=1}^{P}\lambda_i\mu_{ij}a_{ij}}$
M=0; N=0;
for (k=0; k<=P; k++)
{ max_n[k]=0;
for (j=1; j<=R; j++)
{ E(0,ρ_{kj})=1; i=0;
while (E(i,ρ_{kj})>B) {E(i,ρ_{kj})=ρ_{kj}*E(i,ρ_{kj})/(i+1+ρ_{kj}*E(i,ρ_{kj})); i++;}
n[j]=i;
if (n[j]>max_n[k]) {max_n[k]=n[j];}
}
M=M+max_n[k];
}
for (j=1; j<=R; j++)
{ E(0,$\rho_j^{'}$)=1; i=0;
while (E(i,$\rho_j^{'}$)>B) {E(i,$\rho_j^{'}$)=$\rho_j^{'}$*E(i,$\rho_j^{'}$)/(i+1+$\rho_j^{'}$*E(i,$\rho_j^{'}$)); i++;}
n[j]=i;
if (n[j]>N) {N=n[j];}
} End;

Fig. 4. The process of solving utility analytic model.

(2) The utilization of servers

As we all know, most workloads are proportional to their demanded resources [23]. The average resource utilization can be evaluated by the arrival rate of service requests and the serving rate of resources in servers (showed in Equation (8), where parameter b denotes the proportional relation). Thus, the average resource utilization of the dedicated servers and the consolidated servers are illustrated in Equation (9) and (10), respectively. We use Equation (11) to evaluate the ratio in average resource utilizations of the dedicated servers to the consolidated servers ($U_{M/N}$) in our model. It can be easily seen that the exact value of parameter b has no impact on this ratio.

$$U_{ij} = b \times (\lambda_i / (\mu_{ij} \times n_{ij})) \quad (8)$$

$$\overline{U_{M_j}} = \frac{\sum_{i=1}^{P} b \times \lambda_i / (\mu_{ij} \times M_{ij}))}{P} \quad (9)$$

$$\overline{U_{N_j}} = \sum_{i=1}^{P} b \times \lambda_i / (\mu_{ij} \times N_j) \quad (10)$$

$$U_{M/N} = (\frac{\sum_{i=1}^{P} (\lambda_i / (\mu_{ij} \times M_{ij}))}{P}) / (\sum_{i=1}^{P} \lambda_i / (\mu_{ij} \times N_j)) \quad (11)$$

(3) The power consumption of servers

The power consumed by a server over time t is expressed as $E=(S_{base}+(S_{max}-S_{base})\times u_s) \times t$ [1], where S_{base} is the server's baseline power draw, S_{max} is its power draw when serving at maximum capacity and u_s is the average server utilization. In our model, $S_{base\text{-}j}$ is the resource j's baseline power draw in server, and $S_{max\text{-}j}$ is resource j's power draw when serving at maximum capacity. Based on this definition, we can deduce the ratio of power consumption between the dedicated servers and the consolidated servers ($E_{M/N}$) (E_M and E_N denote the power consumed by the dedicated servers and the consolidated servers, respectively) as follows:

$$E_{M_{ij}} = M_{ij} \times S_{base-j} \times t + (S_{\max-j} - S_{base-j}) \times \overline{U_{M_{ij}}} \times t \quad (12)$$

$$E_{N_j} = N_j \times S_{base-j} \times t + (S_{\max-j} - S_{base-j}) \times \overline{U_{N_j}} \times t \quad (13)$$

$$E_M = \sum_{i=1}^{P} \sum_{j=1}^{R} E_{M_{ij}} \quad (14)$$

$$E_N = \sum_{j=1}^{R} E_{N_j} \quad (15)$$

$$E_{M/N} = \frac{E_M}{E_N} \quad (16)$$

4) Application of the utility analytic model

(1) Evaluate on-demand resource allocation algorithms

In our utility analytic model, let M equal N, then we can calculate the relationship of the loss probability (B) in the dedicated servers and in the consolidated servers. Thus, we have the ratio of (1-B) in dedicated servers to that in consolidated servers, which reflects the optimal improvements in QoS (throughput) provided by any on-demand resource allocation algorithms in the consolidated servers compared with in the dedicated servers. The more close the improvements in QoS introduced by an on-demand resource allocation algorithm to such ratio of (1-B), the better this resource allocation algorithm is.

(2) Evaluate virtualization

Our utility analytic model also facilitates evaluating virtualization. Similar to the above evaluation of on-demand resource allocation algorithms, let M equal N, and let all the impact factors of virtualization be 1, then we have the ratio of (1-B) in dedicated servers to that in consolidated servers, which helps to reflect the optimal upper bound of improvements in QoS provided by products of virtualization in the consolidated servers compared with in the dedicated Linux servers in theory.

IV. Performance Evaluation

A. Testbed

We use 17 servers in our experiments, forming a server pool. In the server pool, there are eight servers each of which has two 2.0GHz Quad-Core AMD Opteron(tm) 2350HE processors with 1024KB of cache and 8GB of RAM. We use seven servers to emulate clients of services. The rest two servers are used to be proxy server running LVS [26] and the Besim server for Specweb2005 [27], respectively. The servers are connected with a Gigabit Ethernet.

1) Base System: We use a plain 2.6.18 Linux kernel that comes with the CentOS4.4 standard distribution as our base system for the dedicated servers.

2) Rainbow System: Rainbow is a Xen-based prototype [23] developed by us, which dynamically controls resources allocation among concurrent services via on-demand resource flowing algorithms [22][23]. Xen is a paravirtualization technology that allows multiple guest operating systems to be run in virtual containers (called domains). Each guest OS is a modified version of the base Linux (XenLinux). We use Rainbow system with Xen-3.3.0 for the consolidated servers in our experiments.

B. Experiments Design

The experiments are designed with the goal of verifying the utility analytic model. We consider two typical enterprise services: the Web and database ('DB' for short) services.

- Web service: Apache [18] is used for the Web server. LVS [26] dispatches requests among the Web VMs using round robin (RR) algorithm. SPECWeb2005 [27] is used to generate e-commerce workloads. The average response time is the performance metric.
- DB service: MySQL [16] is used for the DB server. LVS is used to dispatch requests using RR algorithm. We use TPC-W [15] as the DB e-book workloads generator. The size of the DB files is 2.7GB. The DB service is evaluated by the average WIPS (the number of Web Interactions Per Second).

We consider running the Web service and the DB service on dedicated servers and on consolidated servers, respectively. Before comparing the upper bound of physical servers needed to guarantee quality of services hosting on dedicated servers and on consolidated servers, we need to evaluate the overhead caused by Xen to the qualities of those services. We use one physical server, which has two 2.0GHz Quad-Core AMD Opteron(tm) 2350HE processors with 1024KB of cache and 8GB of RAM, to evaluate the degree of such impact on each service caused by one~nine VMs hosted on this server compared to the native Linux. Other works [2][3][4][7][8][9] can also help to determine such impact. Then, based on the above results we verify our model via two groups of experiments.

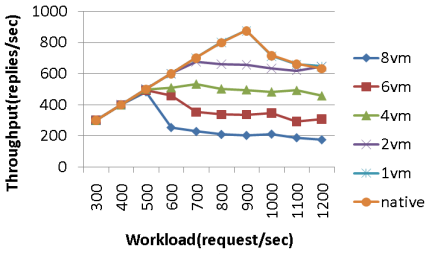
- **Group 1**: We use six dedicated servers, where three servers host Web service and three servers host DB service; while we use two/three/four consolidated servers to host Web and DB services concurrently.
- **Group 2**: We use eight dedicated servers, where four servers host Web service and four servers host DB service; while we use four consolidated servers to host Web service and DB service concurrently.

In all of the above two groups of experiments, on each consolidated server we create two VMs to hosting Web service and DB server, respectively (we call them 'Web VM' and 'DB VM' for short). We allocate six vcpus to each DB VM and pin these vcpus to six physical CPU cores respectively, while we allocate one vcpu to each Web VM. Each VM is allocated 1GB memory. Then the rest CPU cores and memory resources are allocated to Domain 0.

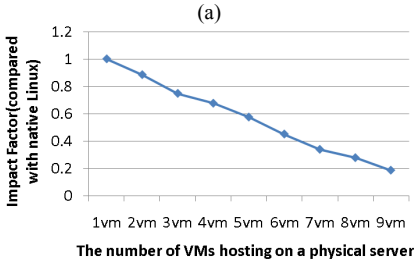
C. Experimental Results

1) Impacts of Xen for Web and DB Services: Using httperf [25] as a generator of Web workloads, we measure the relationship among the request rate (requests/s), the reply rate (replies/s, namely the throughput) and the number of VMs with various file set (illustrated in Figure 5 and Figure 6) in a physical server. In Figure 5 (a) and Figure 6 (a), each curve denotes the relationship between the workloads and throughput of Web service hosted on a native Linux server or on a Xen server with a certain fixed numbers of VMs. Figure 5 shows such relationship when orderly accessing the file set of SPECweb2005 with the size of 5.7GB by requests, in which the disk I/O is the bottleneck resource. All these curves have the same trend: with the increase of workloads the throughput improves first and then degrades, finally remains stable. It is obvious that the throughput degrades with the increase of the number of VMs. We compare the stable mean throughput (corresponding to workloads from 700 to 1200

requests/s) provided by VMs of various numbers with the stable mean throughput (corresponding to workloads from 1100 to 1200 requests/s) provided by the native Linux to calculate the impact factor (*a*) (illustrated by Figure 5 (b)). We sum up the relationship between the impact factor of disk I/O and the number of VMs as a=-0.102v+1.082, where v denotes the number of VMs, using the linear regression. Figure 6 shows such relationship when accessing a file of 8KB by all requests, in which CPU is the bottleneck resource. We find the similar trend in Figure 6 (a) and Figure 5 (a): the throughput degrades with the increase of the number of VMs. From Figure 6 (a) we can also see that the service performance provided by the native Linux is much better than that provided by VMs. Figure 6 (b) shows the impact factors of CPU on various numbers of VM for the Web service. Using linear regression, we sum up the relationship between the impact factor of CPU and the number of VMs as a=-0.039v+0.658, where v denotes the number of VMs.

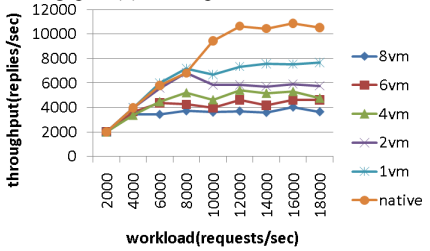


(a)

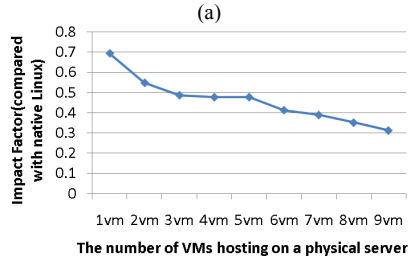


(b)

Fig.5. Requests generated by httperf orderly access the file set of specweb2005 with the total size of 5.1GB. (a) The relationship between Web workloads and throughput. (b) The impact factor of VM for Web service.



(a)



(b)

Fig.6. Requests generated by httperf access a file of 8KB. (a) The relationship between Web workloads and throughput. (b) The impact factor of VM for Web service.

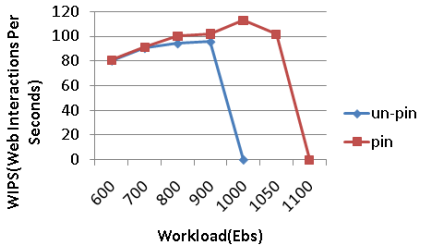
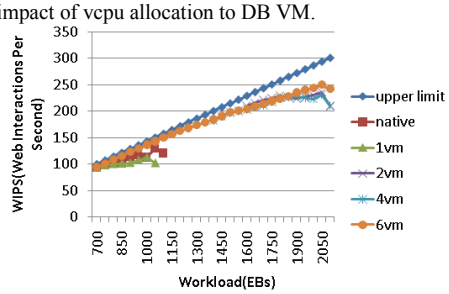
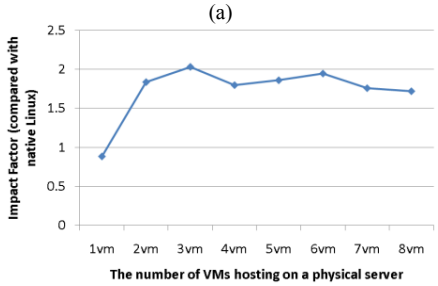


Fig. 7. The impact of vcpu allocation to DB VM.



(a)



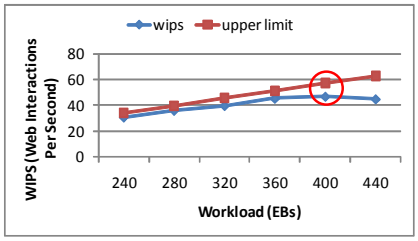
(b)

Fig.8. Requests generated by TPC-W access a 2.7GB database. (a) The relationship between DB workloads and throughput. (b) The impact factor of VM for DB service.

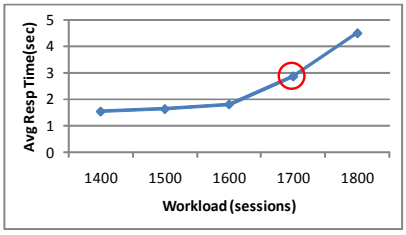
Using TPC-W [15] as a generator of database workloads, we also measure the relationship among the request workloads (EBs—Emulated Browsers), the WIPS (Web Interactions Per Seconds, namely the throughput) and the number of VMs, when requests access a 2.7GB file set (illustrated in Figure 8). Such workload is CPU intensive, thus, this experiment reflects the impact factor of CPU for DB service. During the process of our experiments we find that the allocation of CPU resources to DB VMs has significant impact on the performance of DB service, illustrated in Figure 7. From this figure we can see that pining vcpus of DB VM onto physical CPU cores improves the performance of DB service compared with leaving the scheduling of vcpu to Xen, reflecting the latent room for vcpu scheduling in Xen. Thus, in our following experiments, we allocate six vcpus to a DB VM, and pin each vcpu onto a physical CPU core. At the same time we pin each vcpu of Domain0 onto the rest two physical CPU cores. In Figure 8 (a), each curve denotes the relationship between the workloads and throughput of DB service hosted on a native Linux server or a Xen server with a fixed number of VMs. From this figure we can see that the performance of DB service hosted on the native Linux and one VM is only about half of that hosted on multiple VMs. The reason is that OS software limits the performance improvement for DB service. When multiple VMs (more than one VM) concurrently serving DB requests, CPU, instead of the OS software, is the bottleneck. We sum up the relationship between the impact factor of CPU&software for DB service

(illustrated in Figure 8 (b)) and the number of VMs as $a = \begin{cases} 1.85 & v \geq 2 \\ 0.88 & v = 1 \end{cases}$, where v denotes the number of VMs.

2) Verify the utility analytic model via a case study: We give the following inputs: the number of dedicated servers (M), the Web workloads (λ_w), the DB workloads (λ_d), and the loss probability calculated by requests (B) to verify the model. In order to verify the model reasonably, we select the workloads of each service according to the number of dedicated servers. Figure 9 illustrates the selected workloads signing with red circles. The rule of selecting such workloads is selecting the intensive workload that the servers can afford. The intensive workload means that more or fewer workloads result in remarkable difference compared to it in service performance. Thus, it is clear to see the difference of service performance between services' running on dedicated servers and services' running on consolidated servers. We also measure the serving rate of the bottleneck resource in a physical server for these services using SPECweb2005 and TPC-W. Based on the above results of these experiments, as well as the selected workloads of services, we gain the inputs to the utility analytic model. We mark Web and DB services with w and d, and mark CPU and I/O resources with c and i. Then, the inputs are as follows: μ_{wi}=420; μ_{dc}=100; μ_{wc}=3360; μ_{di}=∞ (the demand on disk I/O by requests accessing DB service is close to zero); a_{wi}=0.8; a_{dc}=0.9; a_{wc}=0.43. We select the parameter B according to the arrival rate and the number of dedicated servers. Using these inputs we calculate the ratio of M to N using the utility analytic model (illustrated in Table 1).



(a)



(b)

Fig.9. The relationship between service workloads and performance on 4 physical servers for (a) DB service and (b) Web service.

TABLE I
THE INPUTS AND OUTPUT TO UTILITY ANALYTIC MODEL

M	λ_w	λ_d	B	N
4	850	200	0.43	2
6	1250	330	0.43	3
8	1700	400	0.43	4
10	2100	500	0.43	5
…	…	…	…	…

We verify our model via two groups of experiments introduced in part B of this Section. Figures 10-13 illustrate the comparison results between dedicated servers and consolidated servers in the service performance and the power consumption. In Figure 10, the left four bars (we only can see three bars, the rest one corresponds to zero, which reflects the failure of this experiment because of too many workloads for servers to afford) denote the performance of DB service running on three dedicated servers, two consolidated servers, three consolidated servers, and four consolidated servers. From these bars we can see that the performance of DB service running on three dedicated servers is the closest to that running on three consolidated servers. The right four bars denote the performance of Web service running on the same server scenarios as the DB service does. From these four bars we can also see that the performance of Web service running on three dedicated servers is the closest to that running on three consolidated servers. This figure helps us draw a conclusion that six dedicated servers should consolidate to three shared servers with the similar service performance in this case study. Figure 11 gives the similar conclusion that eight dedicated servers should consolidate to four shared servers with the similar service performance. At the same time, the average CPU utilization in four consolidated servers improves 1.7 times compared with eight dedicated servers, which is very close to the results (1.5 times) provided by our utility analytic model.

Power Consumption: Figure 12 illustrates the comparisons in the power consumption of eight dedicated servers and of four consolidated servers, when running the service workloads and when idle. The power consumption is measured by an electric parameter tester, which measures the power consumed by one or more servers switching in it. From this figure we can see that consolidated servers save up to 53% power compared with the dedicated servers, providing the similar service performance. The huge savings in the power may result from three reasons. First, the number of consolidated servers reduces 50% compared with the dedicated servers. Figure 12 illustrates that the servers hosting services only increase up to 7% power consumption than the same idle servers. Barroso *et al.*[17] also drew the same conclusion that idle servers cost more than 50% power compared to the busy servers (100% resource utilization). In order to compare the power consumed by the service workloads, we take out the power consumed by idle servers from the total power consumed by the servers hosting service workloads (illustrated in Figure 13). Second, from Figure 13 we can see that the power consumed by the same workloads hosted on consolidated Xen-based servers is 30% less than that hosted on dedicated Linux servers. However, the number of OS is the same. Third, the power consumed by the idle Xen platform is 9% less than that consumed by the same number of idle Linux platform. Why Xen saves more power than the native Linux does? This is a very interesting, open problem. Currently, we have no idea on it because we have no tools to evaluate each component and each instruction contributions to

the total server power. This will be the topic of our future work.

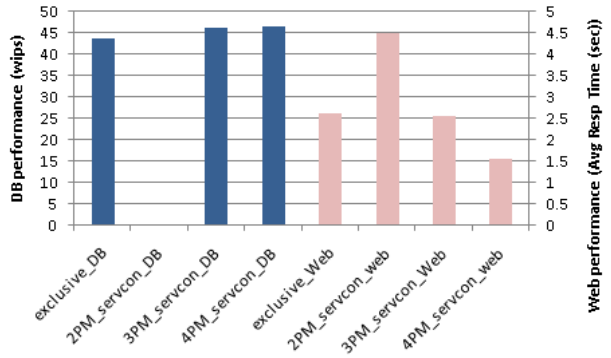


Fig.10. 6 exclusive servers consolidate to *N* shared servers.

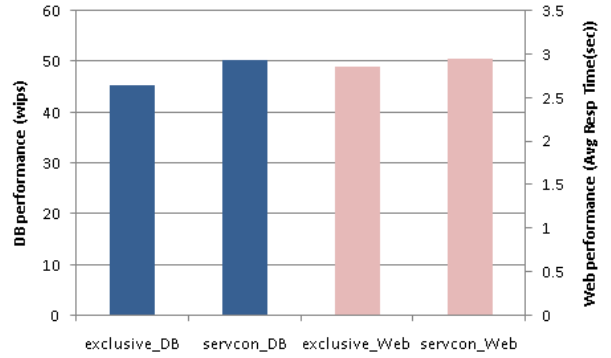


Fig.11. eight exclusive servers consolidate to four shared servers.

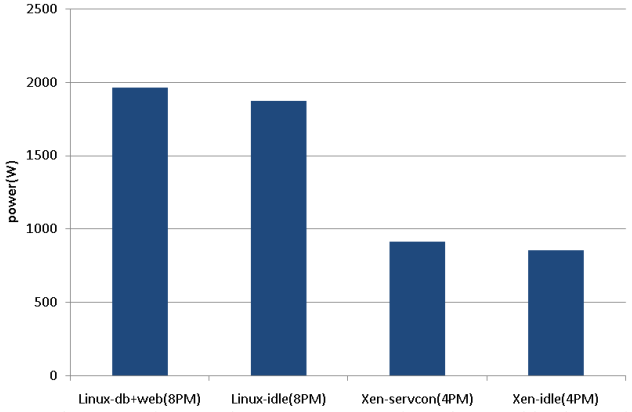


Fig.12. The comparison on the power consumption when workloads running on eight exclusive servers consolidate to four servers. The left two bars correspond to the total power consumption of eight dedicated servers including four servers hosting Web ervice or idle and four servers hosting DB service or idle.

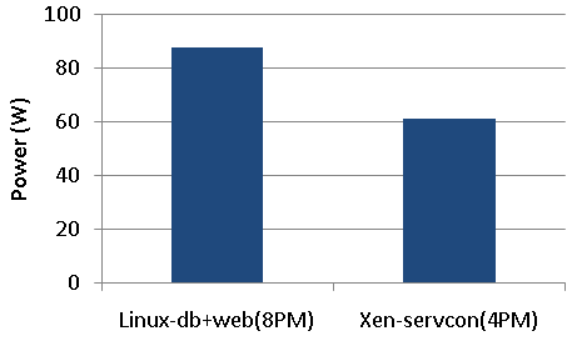


Fig.13. The power consumed by workloads. The left bar corresponds to the total power consumption of worloads on eight dedicated servers including web workloads and DB workloads.

D. Discussions

During the process of experiments, we find two interesting problems.

First, we use a physical server which has two 2.0GHz Quad-Core AMD Opteron(tm) 2350HE processors with 512KB of L2 cache per processor and 8GB of RAM, and a physical server which has two 2.33GHz Quad-Core Intel® Xeon®CPU E5410 with 256KB L1 cache per core and 12MB of L2 cache per processor and 8GB of RAM to host database services with the same e-book workloads. The experimental results show that server which has AMD's 2.0GHz processors improves about 20% throughput compared with server which has Intel's 2.3GHz processors when running the same e-book workloads of database services. The huge difference between the performance of AMD server and Intel server implies that the heterogeneous servers, including servers with different capacities produced by the same manufacturer and servers produced by different manufacturers, must be taken into account. This paper only focuses on the homogeneous/heterogeneous servers produced by the same manufacturer, such as AMD. However, in the current data centers, such as Google, there are many heterogeneous servers with different capacities or produced by different manufacturers. Expanding the utility analytic model to fit data centers with such heterogeneous servers is our future work.

Second, from Figure 5 we can also see that the overhead of Xen on disk I/O is huge, especially when the number of VMs is more than six (the degradation of throughput is more than 50%). This experimental result implies that there's much potential for virtualization to lower its overhead on disk I/O.

V. Conclusion and Future Work

This work proposes and validates a utility analytic model to guide the planning on the scale of Internet-oriented data centers and to analyze the power consumption when using VM-based consolidated servers instead of dedicated ones. Based on the Erlang's loss formula in queuing theory, our model predicts the scale of data centers and the savings in the power consumption before really running the concurrent services, when workloads of these services are consolidated to the VM-based data centers. The experimental results partly validate our model, which show that server consolidation saves up to 50% physical infrastructure, up to 53% power consumption, and improves 1.7 times in CPU resource utilization without any degradation of QoS compared to the traditional dedicated servers in the case study.

In the future, we will focus on expanding the utility analytic model to fit data centers with heterogeneous servers produced by different manufacturers. It is worth noting that the power evaluation of our model cannot be validated because of the lack of tools to measure the power consumption on each component and each request in physical servers. Our experimental results also show that the power consumed by the same workloads hosted on consolidated Xen-based servers is 30% less than that hosted on dedicated Linux servers, at the same time, the power consumed by the idle Xen platform is 9% less than that consumed by the same number of idle Linux

platform. The reason why Xen is better than the native Linux in saving the power is still open.

ACKNOWLEDGMENT

This work was supported in part by the National High-Tech Research and Development Program (863) of China under grants 2007AA01Z119, 2009AA01Z141, 2009AA01Z151, and 2006AA01Z109, and the projects of NSFC under grants 90718040.

REFERENCES

[1] S. Nedevshi, S. Ratnasamy and J. Padhye, "Hot Data Centers vs. Cool Peers", Hotpower08.

[2] P. Padala, X. Zhu, Z. Wang, etc., "Performance Evaluation of Virtualization Technologies for Server Consolidation", HPL-2007-59R1. http://www.hpl.hp.com/techreports/2007/HPL-2007-59R1.html

[3] A. Menon, J. R. Santos, Y. Turner, G. Janakiraman, and W. Zwaenepoel, "Diagnosing Performance Overheads in the Xen Virutal Machine Environment", VEE'05,2005. p.13-23.

[4] D. Gupta, R. Gardner, and L. Cherkasova, "XenMon: QoS Monitoring and Performance Profiling Tool", Technical Report HPL-2005-187, HP Labs, Oct 2005.

[5] S. Govindan, A. Nath, A. Das, etc., "Xen and Co: Communication-aware CPU scheduling for consolidated Xen-based hosting platforms", VEE'07. p.126-136.

[6] S. Mehta, A. Neogi, "ReCon: A Tool to Recommend Dynamic Server Consolidation in multi-Cluster Data Centers", Network Operations and Management Symposium, April 2008, p.363-370.

[7] F. Benevenuto, C. Fernandes, etc., "Performance Models for Virtualized Applications", ISPA 2006, LNCS 4331, p.427-439.

[8] D. A. Menasce, L. W. Dowdy, and V. A. F. Almeida, "Performance by Design: Computer Capacity Planning By Example", Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.

[9] D. Menascé, "Virtualization: Concetps, Applications, and Performance Modeling", In Proc. Of The Computer Measurement Group's 2005 International Conference, Orlando, FL, USA, Dec 2005. p.407-414.

[10] K. H. Yeung, C. W. Szeto, "On the Modeling of WWW Request Arrivals", Proceedings of the 1999 International Workshops on Parallel Processing, 1999, p.248-253.

[11] V. Paxon and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling", IEEE/ACM Transactions on Networking, vol.3, no.3, pp.226-244, 1995.

[12] S. Srikantaiah, A. Kansal, F. Zhao, "Energy Aware Consolidation for Cloud Computing", USENIX Workshop on Power Aware Computing and Systems in Conjunction with OSDI, San Diego, Dec. 2008. p.1-5.

[13] J. Torres, D. Carrera, etc., "Reducing Wasted Resources to Help Achieve Green Data Centers", IPDPS 08, p.1-8.

[14] G. Chen, W. He, etc., "Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services", 5th USENIX Symposium on Networked Systems Design and Implementation (NSDI'08), 2008, p.337-350.

[15] H. W. Cain, R. Rajwar, M. Marden, etc., "An architectural evaluation of java TPC-W", HPCA, 2001, p.229-240.

[16] P. Dubois, "MySQL", NewRiders, ISBN 0735709211, Dec 1999.

[17] L. A. Barroso, U. Hölzle, "The case for energy-proportional computing", IEEE Computer, Vol.40, No.12, Dec. 2007, p.33-37.

[18] R. T. Fielding, G. Kaiser, "The Apache HTTP Server Project", IEEE Internet Computing, vol.1, no.4, July 1997, p.88-90.

[19] X. Wang, etc., "A Resource Management Framework for Multi-tier Service Delivery in Autonomic Virtualized Environments", IEEE Network Operations and Management Symposium, 2008, p.310-316.

[20] G. Jung, etc., "Generating Adaptation Policies for Multi-Tier Applications in Consolidated Server Environments", ICAC08, p.23-32.

[21] P. Padala, X. Zhu, etc., "Adaptive Control of Virtualized Resources in Utility Computing Environments", EuroSys'07, p.289-302.

[22] Y. Song, Y. Li, H. Wang, Y. Zhang, B. Feng, H. Zang, and Y. Sun, "A Service-Oriented Priority-Based Resource Scheduling Scheme for Virtualized Utility Computing", International Conference on High Performance Computing (HiPC), 2008, p.220-231.

[23] Y. Song, H. Wang, Y. Li, B. Feng, and Y. Sun, "Multi-Tiered On-Demand Resource Scheduling for VM-Based Data Center", 9th IEEE International Symposium on Cluster Computing and the Grid (CCGrid), May 18-21, 2009, p.148-155.

[24] M. Armbrst, etc., "Above the Clouds: A Berkeley View of Cloud Computing", Technical Report No. UCB/EECS-2009-28, Feb 2009. http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf.

[25] http://www.hpl.hp.com/research/linux/httperf/

[26] http://www.linuxvirtualserver.org/

[27] http://www.spec/org/web2005/

[28] Y. Chen, A. Das, etc., "Managing Server Energy and Operational Costs in Hosting Centers", SIGMETRICS'05, June, Canada, p.303-314.

[29] J. Chase, D. Anderson, etc., "Managing Energy and Server Resources in Hosting Centers", SOSP, Oct. 2001. p.103-116.

[30] E. Pinheiro, R. Bianchini, etc., "Dynamic Cluster Reconfiguration for Power and Performance", Compilers and Operating Systems for Low Power, Kluwer Academic Publishers, Aug. 2003. p.75-93

[31] K. Rajamani and C. Lefurgy, "On Evaluating Request-Distribution Schemes for Saving Energy in Server Clusters", In Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software, Mach 2003. p.111-122.

[32] E. N. Elnozahy, M. Kistler, R. Rajamony, "Energy Conservation Policies for Web Servers", In Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems, March 2003. p.8.

[33] E. N. Elnozahy, M. Kistler, R. Rajamony, "Energy-Efficient Server Clusters", In Proceedings of the 2nd Workshop on Power-Aware Computing Systems, Feb.2002. p.179-196.

[34] S. Gurumurthi, A. Sivasubramaniam, etc., "DRPM: Dynamic Speed Control for Power Management in Server Class Disks", ISCA 2003. p.169-179

[35] Q. Zhu, A. Shankar, Y. Zhou, "PB-LRU: A Self-Tuning Power Aware Storage Cache Replacement Algorithm for Conserving Disk Energy", In Proceedings of the 18th International Conference on Supercomputing, June 2004. p.79-88.

[36] R. Nathuji, K. Schwan, "VirtualPower: Coordinated Power Management in Virtualized Enterprise Systems", SOSP07, Oct.2007, p.265-278.

[37] D. Gupta, S. lee, etc., "Difference Engine: Harnessing Memory Redundancy in Virtual Machines", OSDI'08, p.309-322.

[38] P. Willmann, S. Rixner, etc., "Protection Strategies for Direct Access to Virtualized I/O Devices", 2008 USENIX Annual Technical Conference, p.15-28.

[39] N. Bobroff, A. Kochut, etc., "Dynamic Placement of Virtual Machines for Managing SLA Violations", IM'07, p.119-128.

[40] T. Wood, P. Shenoy, etc., "Black-box and Gray-box Strategies for Virtual Machine Migration", NSDI'07, p.229-242.

[41] D. Gross, C. M. Harris, "Fundamentals of Queueing Theory (Third Edition)", A Wiley-Interscience Publication, New York, N.Y.10158-0012, p.80-81.

[42] F. Hermenier, X. Lorca, etc., "Entropy: a Consolidation Manager for Clusters", VEE'09, p.41-50.

[43] P. Ranganathan, P.Leech, D.Irwin, and J.Chase, "Ensemble-level Power Management for Dense Blade Servers", ISCA'06, p.66-77.