Abstract—Since the raising of the cloud computing, the applications of web service have been extended rapidly. However, the data centers of cloud computing also cause the problem of power consumption and the resources usually have not been used effectively. Decreasing the power consumption and enhancing resource utilization become main issues in cloud computing environment. In this paper, we propose a method, called MBFDP (modified best fit decreasing packing), to decrease power consumption and enhance resource utilization of cloud computing servers. From the results of experiments, the proposed solution can reduce power consumption effectively and enhance the utilization of resources of servers.

Index Terms—Cloud computing, energy saving, packing problem, power consumption.

1. Introduction

In recent years, the development of multi-core technology leads to the capacity of physical host becoming more powerful. In the past, providing services on web needs many physical hosts to furnish. The resources of hosts usually cannot be used effectively, causing an increase of hardware and maintenance costs. In the cloud computing environment, the resources of each service could be assigned according to the demands of the user. Cloud computing is actually a new concept of network computing, which is composed of multiple or large cluster data centers.

Users can connect to the “Cloud” over the Internet and data centers will handle service requests from users. This makes users feel services spread far and wide. “Cloud” not only affects individuals, but also enterprises and academic enormously. In cloud computing environment, a user can use various services and does not need to know the operating mechanism of the back-end system. This characteristic of cloud computing will slim the client to diminish the cost of maintenance. For developers, it also minimizes the scope of management and burden.

According to McKinsey & Company research report, a company with 200 employees will decrease almost 30% cost only in the part of the software if the maintenance of data centers, the network management, and the software upgrades of this company are operated by “Cloud”.

The cloud computing has been paid attention to globally, users can deal with a lot of computing via the cloud computing. Many large data centers providing cloud services have been appearing. However, there are probably 70% of resources are not used in the data center. In the economic and environmental permission, how to improve the utilization of the data center resource and reduce the energy consumption have become important issues.

Cloud computing possesses the availability and the scalability, however, due to the multi-level, distributed, and virtualized systems, the efficiency of energy will face with more challenges.

About the energy consumption of data center, a research report indicates that costs of a server with 300 watts which runs an entire year is about 338 U.S. dollars, and produces 1300 kg of carbon dioxide [1]. According to the research report in 2006, these basic infrastructures consumed 450 million U.S. dollars, which equate to 1.5% of total U.S. energy consumption. It also derives many problems, such as increasing the cost, wasting electricity, and emitting more carbon dioxide. It not only affects the development of enterprises, but also causes the damage to the environment [2].

2. Related Work

2.1 Cloud Computing

The concept of “Cloud” is referred the network diagram, because the engineers often take a cloud icon to represent the “Network” in drawing the diagram. Cloud computing is the network concept of “as a service”, and it provides various services for users with the large number of service nodes. The cloud providers use the virtualization technique to build and serve for users with diversified services. Therefore, no matter the user requests the network storage or requires the computing ability of software, they can be achieved through the cloud providers to reduce the cost of the hardware and the software.

There are three tiers of the industry of cloud computing:
1) Software as a service (SaaS): it does no longer have to confine the differences of platform. Users can use services in the cloud to replace the software installed in local.

2) Platform as a service (PaaS): the user can develop, install, integrate, and execute a variety of applications and information systems on a platform as the service of the cloud. They could program and install services in the cloud through the network.

3) Infrastructure as a service (IaaS): the cloud provider offers various types of services, like web server, data storage device, CPU capability, and network equipment. They could integrate those services as infrastructure such as IT systems and databases to rent to the company without the ability to purchase and manage that equipment.

The cloud computing can be classified into four categories by supplying services as follows.

1) Public cloud: a service provider supplies the construction, maintenance, operation, and management of the cloud environment, and multiple businesses and users share this cloud. The public cloud services which open to common users through the network and third-party service provider, but “public” does not mean “free”, it may also represent a free or fairly inexpensive. The public cloud does not indicate that user’s data is available to anyone; the public cloud providers usually implement access control mechanisms to individual users.

2) Private cloud: many merits of the public cloud are also in the private cloud, such as flexibility and suitability for providing the service. The difference between them is that the information and procedures are managed by the organization in the private cloud, and it is not subject to network bandwidth, security concerns, and regulatory constraints as that in the public cloud. In addition, due to the user and the network are subject to special restrictions, the providers and users own more control about the cloud infrastructure, the system safety and flexibility can be improved in the private cloud.

3) Community cloud: the community cloud is controlled and used by many organizations which own the similar interests, such as specific security requirements and common purposes. Community members can share cloud data and applications.

4) Hybrid cloud: hybrid clouds combine the public cloud and the private cloud. In this model, users usually outsource non-business critical information processing in the public cloud and control of business-critical services and information.

2.2 Virtualization

Virtualization is a catalyst for the cloud computing. “Virtualization” appeared in the 1960s first, it is the technology developed by IBM Corporation in order to allow users fully use of the expensive mainframe resources. By virtualization technology, the original mainframe can be divided into a number of logic cells, i.e., virtual machines (VMs), and each VM can run on the same host. Fig. 1 shows the schematic diagram of the virtualization architecture[3][4].

The concept of virtualization technology is a reproduction of the physical machine, which can non-directly relate to the hardware in using, and provide the machine architectures according to users’ requirements virtually. In general, VM will improve the physical machine compatibility constraints, so as to overcome the hardware resource limits and provide higher adaptability.

The virtualization technology can be divided into two categories by serving targets: system virtualization and process virtualization. The difference between them is the scope of virtualization. The system virtualization provides a complete system platform and performs an independent guest operating system (OS). The process virtualization provides services only on a single process.

The virtualization technology can be divided into two categories by serving targets: system virtualization and process virtualization. The difference between them is the scope of virtualization. The system virtualization provides a complete system platform and performs an independent guest operating system (OS). The process virtualization provides services only on a single process.

The general situation of the system on the physical machine is divided into several layers, which contains the underlying hardware resources, OS, hardware drivers, and applications. Being able to run more than one VM on a physical machine, it is necessary to add a software layer in the middle of the OS layer and the bottom hardware resources layer, which names the “hypervisor” or “VM manager” (VMM), to manage hardware resources and guest OS scheduling. Fig. 2 shows the VMM and hypervisor architecture.

A. Characteristics of Virtual Platform

The common virtual platform in present includes: VMware Server, Xen, and KVM.

VMware Server is a commercial virtual platform and the architecture is the OS-hosted VMMs type. It mainly supports the full-virtualization and can work on Windows.
and Linux OS. Xen is an open source (GNU/GPL license) virtual platform to work directly on the physical hardware of the hybrid VMMs type. It mainly supports paravirtualization. KVM is an open source virtual platform which supports x86 OS virtualization. Since KVM is implemented as a loadable kernel module, its architecture complies with the OS-hosted VMMs.

Excepting VMware Server, Xen, and KVM, QEMU virtualization simulates the hardware device by translating the binary code of the instruction code block. When QEMU receives the command, it will interpret the command line by line, and the interpreted command group will be stored in the buffer of blocks. If the same command is executed, QEMU will jump to the block directly to execute without interpreting the command again so as to accelerate the speed of processing instructions and the simulation speed.

2.3 Virtual Machine Migration

The migration is an important function in virtualization. VM migration is to move the VM and applications executing on the original host to the target host.

Through the system migration technology provided by the virtual platform, each VM is treated as a set of units that can be completely moved. Migrated VMs do not provide the updates of target machine software and hardware resources after the migration completed [5][6].

The migrated VM can not only be managed in a unified interface, but through some VM software, such as KVM, it provides high-availability tools. If any VM crashes due to unpredictable reasons, it could automatically switch to the other virtual server in the same network domain to work continuously without the interruption.

The advantage of migration is to simplify system maintenance and improve the balance of system loading. In other hand, the migration enhances the fault tolerance and optimizes the power management of whole system.

Migration can be classified into two categories, the online migration and the offline migration. The differences between them are as follows:

- Offline migration: it must shut down or suspend the VM before the migration, and restart or continue the implementation after the migration. During the migrating processes, the client will not be able to access services provided by this VM.
- Online migration: there is needless to shut down or suspend the VM in migrating, and the client can continue to use the services provided by this VM during the migration processes.

The benefit of offline migration is faster in total migration time, but the VM must suspend during the migration. Before the migration completed, the VM is unable to provide any service.

Although it takes longer time of the online migration, the executing VM could migrate to another physical machine without suspending, and the user does not feel the interruption in the process.

The typical online migration behavior can be divided into three stages:

1) Push phase: the VM of source node is still running and system contents are moving simultaneously.
2) Stop-and-copy phase: the VM of the source node has stopped and the moving of system contents is in the final stage. The VM of destination node will begin to boot after migrating.
3) Pull phase: the VM migrated to the target node has executed. If the content accessed by the system is not copied from the source node, it will proceed to send this lost content from the source node.

2.4 Dynamic Programming

In addition to the greedy algorithm, “dynamic programming” is a common and effective method to solve the optimization problem. In the dynamic programming, a solution to the problem could be deemed to a result of a series of choices. It needs to check each making-decision sequence, which contains one of the best subsequence to ensure that the finally obtained solution is one of the global optimal solutions in the dynamic programming.

Dynamic programming algorithms can be divided into three steps:

1) Find an optimal solution and its structural characteristics, and then define the value of an optimal solution recursively.
2) The bottom-up recording the calculated numerical way to calculate the value of an optimal solution.
3) To construct an optimal solution of the structure from the recorded information.

The packing problem is a dynamic programming algorithm to solve optimization problems. It starts from an empty set, after adding an element into this set, calculating the optimal solution of this stage until all elements are added into this collection. The end result will be the optimal solution.

2.5 Wake-On-LAN

The wake on LAN (WOL) is a technique to switch the dormant or shutdown computer to operating or power-on status over the network. The Advanced Manageability Alliance which IBM proposed first in April 1997 is followed by other businesses and industries gradually.

The motherboard and the network card on the shutdown or dormant computer still retain the weak power supply, and this weak power supply of the network card is enough to maintain the capability of the minimum operation. The network card could receive data from the broadcast, and receive information to interpret and detect. If the network card receives the specific broadcast content, it will interpret and response. This particular packet is called “the magic packet”. If the magic packet is destined to the local address, it will notify the motherboard and power supply to proceed the turn-on or wake-up procedure.

The WOL needs to achieve the following five requirements:
Fig. 3. Concept of energy-saving policy.

1) The motherboard supports the PCI 2.2 expansion slot.
2) The power management options in the BIOS (basic input/output system) are set correctly.
3) The network card supports the PCI 2.2 standard.
4) An ATX type power supply.
5) The software supports WOL.

6) An ATX power supply with the correct BIOS power management settings will support the weak power for the motherboard and network card by the PCI bus in the dormant or shutdown computer.

3. System Design and Processes

With the rising of cloud computing, many large data centers to provide cloud computing services have appeared. Now how to improve the energy efficiency under the economic and environmental conditions is becoming an important issue. Generally, CPU consumes about 40% energy when a computer is running. It means this 40% energy consumption can be saved if we shut down a computer [7]. Thus, if we can gather those virtual devices in some servers and turn off some others, we could improve the resource usage and lower the power consumption to save energy, as shown in Fig. 3.

3.1 System Architecture

A modified best fit decreasing packing (MBFDP) mechanism has been proposed in this paper to integrate servers of cloud computing for energy saving.

In order to sort hosts and VMs, it needs a value to represent the score of the host and VM. In the packing problem, the volume and the weight of a box can be carried are limited, and the score of a box is calculated by the weight and volume. Similarly, in the cloud computing environment, the CPU and memory of each physical machine are limited. In the proposed method, based on the packing problem, a two-dimensional object scores [8]:

\[
S_j = \lambda W_j + (1 - \lambda) V_j \quad (j \in N)
\]

is used for calculating the scores of hosts (Eq. 2) and VMs (Eq. 3). The hosts are ordered by the score from the smallest to the largest to find the node with enough capacity for migration. The VM with the largest score is migrated first. In this way, the total number of migrations can be reduced significantly.

\[
S_j = \frac{\text{The num of CPU assigned for VM}}{\text{Total num of physical CPU}}
\]

\[
W_j = \frac{\text{The memory size assigned for VM}}{\text{Total physical memory size}}
\]

\[
\text{Host score} = P \cdot cu_j + (1 - P) mu_j \quad (2)
\]

\[
\text{VM score} = V \cdot vc_j + (1 - V) vm_j \quad (3)
\]

where

\[
P = cu_j / (cu_j + mu_j), \quad \text{and}
\]

\[
V = vc_j / (vc_j + vm_j).
\]

The excessive frequency of switching may cause quickly reducing the life-span of the server. If there is a lot of work to join and complete in a short time, those shut down servers will be turned on immediately, thus, the automatic integration of VM will result in great waste of energy. In order to further reduce energy consumption, we use a time series prediction model to reduce the switching frequency of the server for saving energy.

The prediction model based on the interval of each hour records the historical data of services demands to obtain the peak value and predicts the future number of hosts required by using the peak value and the capability of the host in MIPS (million instructions per second). The capacity of each VM is simulated by the requirements of the work loading. The allocation of CPU and virtual CPU (VCPU) is based on one-to-one configuration. The required number of physical hosts in each interval is defined as the historical requirement and can be calculated by

\[
\text{Historical requirement} = \frac{\text{Peak value}}{C}. \quad (4)
\]

In (4), the peak value is the summa of the peak MIPS of each VM executing in an interval in historical data, and C is the capability of a physical host.

3.2 System Processes

The proposed dynamic energy-saving policy of cloud servers is to integrate VMs for saving energy by the minimum number of host support.

Fig. 4 shows the flow chart of the proposed mechanism. In the figure, T is a period of time, Run_num represents the
number of machines turned on currently, Require_num is the number of machines required currently, and All_machine_state represents the current state of the machine. There are three states of the machine: Idle, Active, and Dormant. Id_machine_num represents the number of the machines running but not working currently. Ac_machine_num is the number of the machines operating and working currently. Dormant means the dormant state of the machine. Queue is a waiting queue. The process is described as follows:

1) Set $T$, Run_num, and Require_num initially and go to Step 2.
2) Monitoring Run_num, Require_num, and All_machine_state, go to Step 3.
3) Checking if Run_num is greater than Require_num, then go to Step 5, otherwise go to Step 4.
4) Checking if Run_num is less than Require_num, then go to Step 8, otherwise go to Step 10.
5) Determining if Ac_machine_num is greater than Require_num, then go to Step 7, otherwise go to Step 6.
6) Calculating the number of machines can be dormant by Run_num minus Require_num, and via WOL to make those machines into the dormant state, then go to Step 10.

Fig. 4. Energy-saving flow chart.
### Table 1: Hardware architecture of the experiment environment

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel® Core™2 Quad Q9500 2.83 GHz</td>
</tr>
<tr>
<td>Memory</td>
<td>DDR3 1333 8 GB</td>
</tr>
<tr>
<td>Hard disc</td>
<td>SATAII 500 GB</td>
</tr>
<tr>
<td>UPS</td>
<td>320 W</td>
</tr>
</tbody>
</table>

### Table 2: Software architecture of the experiment environment

<table>
<thead>
<tr>
<th>Software</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>Microsoft Windows 7</td>
</tr>
<tr>
<td>Experimental</td>
<td>CloudSim</td>
</tr>
<tr>
<td>IDE</td>
<td>Eclipse</td>
</tr>
</tbody>
</table>

### Table 3: Total number of migrations

<table>
<thead>
<tr>
<th>Works</th>
<th>BFDP</th>
<th>MBFDP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
<td>150</td>
</tr>
<tr>
<td>1st time</td>
<td>102</td>
<td>151</td>
</tr>
<tr>
<td>2nd time</td>
<td>100</td>
<td>151</td>
</tr>
<tr>
<td>3rd time</td>
<td>103</td>
<td>147</td>
</tr>
<tr>
<td>4th time</td>
<td>100</td>
<td>142</td>
</tr>
<tr>
<td>5th time</td>
<td>105</td>
<td>151</td>
</tr>
<tr>
<td>Avg.</td>
<td>102</td>
<td>148.4</td>
</tr>
</tbody>
</table>

### Table 4: Energy consumption of BFDP and MBFDP

<table>
<thead>
<tr>
<th>Works</th>
<th>BFDP (kWh)</th>
<th>MBFDP (kWh)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st test</td>
<td>4.51</td>
<td>4.52</td>
</tr>
<tr>
<td>2nd test</td>
<td>4.49</td>
<td>4.47</td>
</tr>
<tr>
<td>3rd test</td>
<td>4.50</td>
<td>4.48</td>
</tr>
<tr>
<td>4th test</td>
<td>4.46</td>
<td>4.44</td>
</tr>
<tr>
<td>5th test</td>
<td>4.53</td>
<td>4.51</td>
</tr>
<tr>
<td>Avg.</td>
<td>4.50</td>
<td>4.49</td>
</tr>
</tbody>
</table>

### Fig. 5. Relationship of works and migrations.

4. Results and Analysis

In the simulations, the host computer is equipped with a 2.83 GHz Intel® Core™ 2 Quad CPU, 8 G DDR3 memory, 500 G SATAII hard disk, and 320 W UPS. We take CloudSim as our experimental platform. CloudSim is a cloud environment simulation platform developed by the University of Melbourne, Australia, programmed in Java language. The OS on the host is Microsoft Windows 7 Enterprise. The experimental environment is shown in Table 1 and Table 2.

Due to CloudSim is programmed by Java language, we need to use Java IDE (integrated development environment) for coding, and we take Eclipse as Java IDE. The Eclipse IDE is not only a tool to write, compile, debug, and deploy programs written in Java, but can support other programming languages.

The experiments are divided into two parts:

Experiment 1 is to measure the total number of migrations and the situation of energy consumption by applying the proposed MBFDP and the best fit decreasing packing (BFDP) mechanisms respectively under a variety of works.

Experiment 2 is to simulate the utilization of resources during each period in a small scale cloud data center in one day for comparing the different results by applying the proposed integrated energy-saving mechanism and only applying the MBFDP algorithm.
Table 5: Energy consumption during each period

<table>
<thead>
<tr>
<th>Time</th>
<th>MBFDP + Predict</th>
<th>MBFDP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Energy consumption (kWh)</td>
<td>Energy consumption (kWh)</td>
</tr>
<tr>
<td>08:00–09:00</td>
<td>1.58</td>
<td>2.43</td>
</tr>
<tr>
<td>09:00–10:00</td>
<td>2.03</td>
<td>3.06</td>
</tr>
<tr>
<td>10:00–11:00</td>
<td>2.59</td>
<td>3.07</td>
</tr>
<tr>
<td>11:00–12:00</td>
<td>3.25</td>
<td>3.03</td>
</tr>
<tr>
<td>12:00–13:00</td>
<td>3.44</td>
<td>3.59</td>
</tr>
<tr>
<td>13:00–14:00</td>
<td>3.48</td>
<td>3.37</td>
</tr>
<tr>
<td>14:00–15:00</td>
<td>3.46</td>
<td>3.38</td>
</tr>
<tr>
<td>15:00–16:00</td>
<td>3.05</td>
<td>3.03</td>
</tr>
<tr>
<td>16:00–17:00</td>
<td>4.40</td>
<td>3.94</td>
</tr>
<tr>
<td>17:00–18:00</td>
<td>3.62</td>
<td>3.63</td>
</tr>
<tr>
<td>18:00–19:00</td>
<td>4.29</td>
<td>3.96</td>
</tr>
<tr>
<td>19:00–20:00</td>
<td>3.64</td>
<td>3.57</td>
</tr>
</tbody>
</table>

Fig. 7. Energy consumption diagram.

The prediction model can reduce averagely 2% of the total energy consumption than MBFDP without time prediction. Therefore, the effective prediction could reduce the energy consumption and decrease the switching frequency of the server. It increases the validity of the hardware to achieve better energy-saving effect.

5. Conclusions

Applying the VMs migration can realize the dynamic reallocation of resources and reduce energy consumption effectively in the cloud computing environment. In this paper, we propose an integrated energy-saving mechanism: MBFDP. Comparing with the typical algorithms of packing problems: BFDP, the proposed mechanism can reduce the number of hosts required and reduce the total number of VMs migrations effectively.

The experimental results show that by comparing the MBFDP with BFDP mechanisms, the former can reduce the total energy consumption by 12% to 14% since the order of migration in BFDP will produce the additional energy consumption caused by the extra number of migrations.

In this paper, we also propose a dynamic energy-saving mechanism for cloud servers based on MBFDP integrated with the time prediction model, which centralizes VMs as possible to achieve the minimum number of host required for saving energy.

We simulated a small cloud data center for comparing the integrated dynamic energy saving mechanism and the MBFDP mechanism. The results show that the former mechanism can reduce 2% of the total energy consumption averagely. Therefore, a better energy saving can be achieved effectively by applying the proposed integrated dynamic energy-saving mechanism.

References


Han-Sheng Chuang received the B.S. degree from the Chang Jung Christian University in 2004 and the M.S. degree from the Yu-Da University in 2009, both in information management. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Tatung University. His research interests include wireless network and QoS of next generation telecommunication.

Liang-Teh Lee received the B.S. degree in electrical engineering from Tatung University in 1971 and his M.S. and Ph.D. degrees in computer engineering from University of Southern California in 1978 and 1989, respectively. He was the Director of the Computer Center and the Chairman of the Department of Computer Science and Engineering of Tatung University. Currently, he is a professor with the Department of Computer Science and Engineering, Tatung University. His research interests include computer architectures, parallel and distributed computing, embedded systems, high speed switching architectures, and wireless networks.

Che-Yuan Chang received the B.S. degree from Taipei Chengu University of Science and Technology in 2009 and his M.S. degrees in computer science and engineering from Tatung University in 2012. His research interests include cloud computing and VM resource management.

Chia-Ying Tseng received a bachelor degree in industrial education from Taiwan Normal University in 1979 and a master degree in computer engineering from Arizona State University in 1985. In 1990, he was selected by Ministry of Education to enter the Ph.D. program with the Department of Computer Science and Engineering, Arizona State University for one year research. Currently, he is an assistant professor with the Department of Computer Science and Engineering, Tatung University. His research interests include embedded system design, embedded real time operating system, multiprocessor system-on-chip (MPSoC), and cloud computing and virtualization.