E2CC: Building Energy Efficient ClassCloud using DRBL

Che-Yuan Tu[#1], Wen-Chieh Kuo[#2], Yao-Tsung Wang[#3], Steven Shiau[#4]

[#]*National Center for High-Performance Computing*
No. 7, R&D Rd. VI Hsinchu Science Park, Hsinchu, Taiwan
[1]rider@nchc.org.tw
[2]rock@nchc.org.tw
[3]jazz@nchc.org.tw
[4]steven@nchc.org.tw

Abstract

Green Computing is a growing research topic in recent years. Its goal is to increase energy efficiency and reduced resource consumption. Building infrastructure for Cloud Computing is one of the methods to achieve these goals. The key concept of Cloud Computing is to provide a resource sharing model based on virtualization, distributed filesystem and web services. In this paper, we will propose an energy efficient architecture for Cloud Computing using Diskless Remote Boot in Linux (DRBL) and Xen. We call this architecture ClassCloud because it is suitable to build an experimental cloud infrastructure in PC classrooms. According to our power consumption experiments, the diskless design of DRBL provides a way to implement a power economization computing platform. Computers booting from network without hard disk installed will reduce power consumption from 7% to 33% compared to computers with hard disk installed.

Keywords: *Green Computing, Cloud Computing, Diskless Remote Boot in Linux (DRBL), Xen*

1. Introduction

Cloud Computing is leading a new trend of next generation HPC. This issue also brings a great impact both in IT industries and academia even though no common definition and actual standard exits yet. Ian Foster, the “Father of Grid Computing” made a definition to those already saturated list of definitions for Cloud Computing in the thesis named “Cloud Computing and Grid Computing 360-Degree Compared” [4], and it brings out three key elements which are referred to as computing power, virtualization and storage in our ClassCloud platform. Today, the internet's leading retailer, Amazon.com, provides a new web service called Amazon EC2 which is a kind of business model based on Web 2.0 concept. Users have to pay for their experience in EC2, even if on the trial. For this reason, we all consider that there should be such a tool as the Cloud Computing learning and teaching toolkit for Open source, thus the ClassCloud project is to be carried out for solving such problem.

Education institutions or research organizations that demand to adopt Cloud Computing to solve computing and energy saving problems may use our toolkit for practice or experiment. The ClassCloud infrastructure is composed of DRBL [7] and Xen, each of them acts as deployment tool and virtualization solution. Furthermore, DRBL also helps a lot on energy saving and cost down because of the diskless design. So there is no need to have hard disks in client machines instead the DRBL server handles all of the necessary files of client machines. In Chapter 5, we will show how DRBL really works on power saving and how much DRBL can save. Consequently, Users who want to easily build an essential energy efficient Cloud Computing environment for education or investigation in their PC classrooms may use our toolkit easily for free. On the other hand, most of the PC classrooms are usually used only for teaching and learning, hence lots of computing resource is unused or underutilized. That's the reason why we use DRBL and virtualization technologies to deal with such problem. ClassCloud project helps you to switch your PC classrooms between different purposes easily and rapidly. In this paper, we will introduce our ClassCloud project architecture and tell you how it really works with some special applications.

2. Background

2.1. Green Computing

Green Computing is the environmentally responsible use of computers and related resources [12]. The goal of Green Computing is efficient resource utilization and reduced resource consumption. Some suggested practices for Green Computing are the following ways: (1) find out how much energy in IT system; (2) ensure unused equipments are turned off when it is not being used; (3) educate staff to the benefits of saving energy and recycling; and (4) identify IT management practices that reduce power consumption [3]. Some approaches of Green Computing are algorithmic efficiency, virtualization, terminal server, and power management (include power supply, storage, video card and display) [14].

2.2. DRBL

Diskless Remote Boot in Linux (DRBL) is an open source solution to managing the deployment of the GNU/Linux operating system across many clients [7]. DRBL supports lots of popular GNU/Linux distributions, and it is developed based on diskless and systemless environment for client machines. DRBL uses PXE/Etherboot, DHCP, TFTP, NFS and NIS to provide services to client machines so that it is not necessary to install GNU/Linux on the client hard drives individually [7]. Users just prepare a server machine for DRBL to be installed as a DRBL server, and follow the DRBL installation wizard to configure and dispose the environment for client machines step by step. That's really an easy job to deploy a DRBL environment on clustering systems even for a GNU/Linux beginner, hence cross-platform and user-friendly are the key factors that make the DRBL become a superior clustering tool.

2.3. Xen

Xen is an open source virtual machine monitor (VMM/Hypervisor) for x86, x86_64, IA64, PowerPC, and other CPU architectures [16]. According to paper of Paul Barham and Boris Dragovic et al. [9], Xen uses para-virtualization to avoid performance losses of full virtualization, but the cost is to modify the guest operating system. It makes VM (virtual machine) with near-native performance. Besides, Xen also supports full virtualization through hardware virtualization support (Intel-VT, AMD-V) and offers VM live migration. The Amazon EC2 is currently one of the most well-known solution for utilizing a highly customized version of the Xen hypervisor [1].

3. Architecture overview for ClassCloud

ClassCloud is the project that helps to easily build cloud infrastructure in PC classrooms. It is usually suitable for PC classrooms because of the following reasons: (1) PC classrooms need unified system and central management; (2) it is suitable for highly computing-oriented applications rather than storage-oriented; and (3) the diskless design saves power consumption from hard disk.

PC classrooms administrator uses ClassCloud to provide Cloud Computing infrastructure without influencing original system, because it is builed based on the DRBL architecture. DRBL helps to easily and quickly deploy clustering system for PC classrooms and change system environment without complicated steps. DRBL client is diskless; it means DRBL does not change original system in PC classrooms when clients have local hard disks. For management convenience, DRBL offers central management interface to monitor and control client PCs, so it also helps to save lots of time to manage massive PCs.

Fig. 1. shows the ClassCloud architecture. DRBL server is responsible for PC deployment and management. DRBL clients are in diskless environment, and the root filesystems of DRBL clients are stored on DRBL server. Therefore, non-usage hard disks can be merged into an extra shared storage. For creating virtual cluster, ClassCloud uses virtualization technology to elastically create all kinds of VMs in image-based or diskless. The advantage of image-based VM is that user can create virtual cluster by using their images, and the advantage of diskless VM is their software stacks are the same with DRBL clients. If users want to create virtual clusters, they can just log into DRBL sever and use the DRBL commands to deploy their virtual clusters.

For archiving ClassCloud capability, ClassCloud needs deployment, management mechanism and virtualization technology. Therefore, ClassCloud is based on two components: DRBL and Xen. DRBL can support user to massively deploy and effectively manage clusters. Besides, Xen is an open source virtualization solution, it can logically divide machine resource and increase resource utilization.

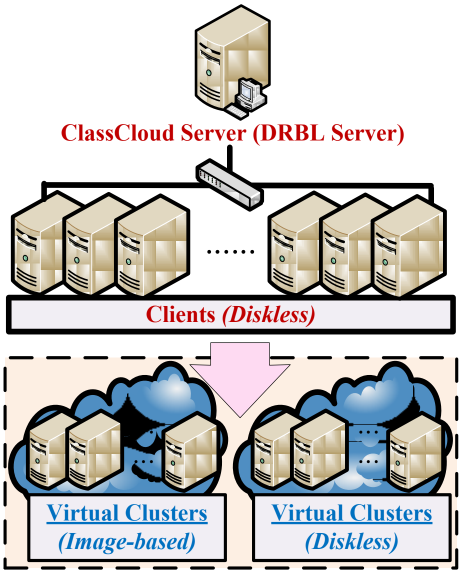

Fig. 1. ClassCloud architecture

Fig. 2. shows the software stacks of ClassCloud. The Linux kernel software stack is usually divided into two layers: (1) Kernel space; and (2) User space. The Xen hypervisor, Linux kernel and kernel modules belong to kernel space. Xen hypervisor is the lowest level, it handles hardware access between host and guest/VM. The Linux kernel is needed to be patched for the support of Xen. The DRBL and Xen tools are in user space. DRBL can automatically bind these Linux daemons (TFTPd, NISd, DHCPd, and NFSd) to build up cluster environment and offer management utilities to control cluster.

Fig. 3. is the operation of ClassCloud. User sends a request to ClassCloud server for creating virtual cluster. After that user chooses appropriate VM image or diskless VM, then the ClassCloud server allocates it to idled clients. ClassCloud server follows user's configuration (number of VMs, VM images, available DRBL clients) to create virtual cluster. Finally, user can run their applications on their own virtual clusters.

4. Implementation

The above chapter describes the ClassCloud architecture, software stacks and operation. This section will explain how to build ClassCloud and deploy massive PCs in PC classrooms. Table 1 is the hardware specifications and software list of ClassCloud.

4.1. Requirement

The bottom layer of ClassCloud (see Fig. 2) is Xen hypervisor. Xen hypervisor uses para-virtualization technology, so it is necessary to modify a Xen supported Linux kernel. After installing Xen hypervisor and Xen patched kernel, the following step is to install Xen tools for easier VM management.

4.2. Deployment

User can install their own necessary packages or software before installing DRBL on the server . The DRBL will use the server system environment to deploy it to DRBL clients in PC classrooms. DRBL has features of simple and fast deployment, and it has two phases of deployment. First phase, DRBL will create customized kernel for clients and install required software (TFTP, NIS, DHCP, and NFS). Second phase, DRBL will ask user the number of PCs and some deployment related information. According to user information, DRBL will automatically setup daemons (TFTP, NIS, DHCP, and NFS) and create root filesystems for DRBL clients on DRBL server. User has to pay attention on the preferences of network boot via PXE, so the setting of Xen PXE is required. After that, DRBL clients also have to setup the BIOS for PXE network booting. Then, if the DRBL clients boot successfully from network via PXE, the deployment of DRBL-Xen cluster is completed.

4.3. Operation

Administrator uses the DRBL management interface to manage and control DRBL clients. When there are computing demands, the ClassCloud can allocate appropriate computing resource to meet the requirements from PC classrooms. Therefore, the ClassCloud server will ask Xen to dynamically create VMs by following user's instructions without influencing original system.

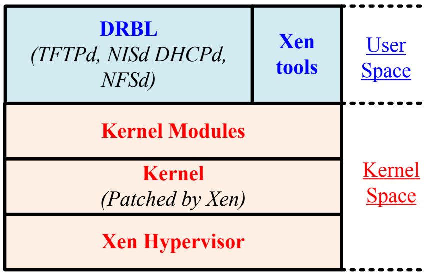

Fig. 2. Software stacks of ClassCloud

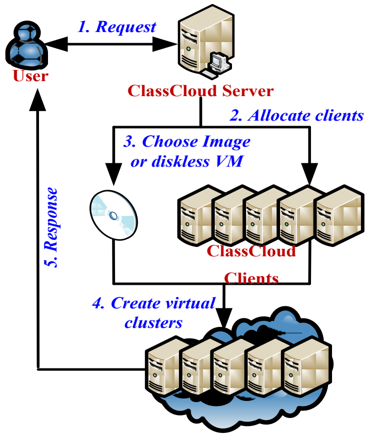

Fig. 3. Operation procedure of ClassCloud

Table 1.

Hardware specifications and Software list

Hardware (PC)	Software
Intel(R) Core(TM)2 Quad CPU Q6600 @ 2.40GHz	Debian GNU/Linux 5.0 (lenny) AMD64
1GB RAM	Kernel 2.6.22.9
160GB Hard Drive	DRBL 1.9.4-27
Intel 82571EB Gigabit NIC	Xen 3.4.0
Hardware (Network switch)	
Linksys SLM2048 48-port 10/100/1000 Gigabit Switch	

4.4. Discussion

ClassCloud helps to quickly and easily build Cloud infrastructure in PC classrooms, and it also provides effective utilization rate and flexible resource allocation for the architecture. The ClassCloud infrastructure is composed of DRBL and Xen, and each of them acts as deployment tool and virtualization solution.

According to users' demands, the ClassCloud helps user to dynamically create appropriate virtual cluster for computing requirement, so user can easily get enough computing resource without complex procedures from it. The ClassCloud is highly suitable for education and science research, and it can quickly deploy related required environment for these domains. Although virtual cluster deployment of ClassCloud is very elastic, the lack of good performance is still a secret concern. Hence, the ClassCloud takes Xen as the high-performance virtualization solution for better efficiency because of the para-virtualization. Table 2 is the comparison between physical and virtual cluster in the ClassCloud.

5. Experiment Design

5.1. Power evaluation experiment of ClassCloud

Fig. 4. is the power evaluation architecture of ClassCloud. This testbed includes one ClassCloud server , twelve ClassCloud clients, one network switch and one electricity monitor for data collection. All of the testbed facilities are powered by the power distribution unit (PDU) except the electricity monitor, and the electricity monitor that connects to PDU is responsible for the data collection of electricity fluctuation in this testbed.

The electricity evaluation test includes three categories which are booting procedure, I/O-bound test and CPU-bound test. All these categories are performed with three evaluation cases: (1) local boot with HD: the ClassCloud clients boot from local hard disks with root filesystem; (2) PXE boot with HD: ClassCloud clients boot from ClassCloud server via PXE instead of booting from local hard disks, but the local hard disks are still power-on. It means that the operating system can detect hard disks without mounting it; and (3) PXE boot without HD: ClassCloud clients boot from ClassCloud server via PXE without hard disks built-in for real diskless environment. The following section will make analysis of the power consumption between the three cases. At the end of analysis, the experiment conclusion will tell the most suitable time for running dedicated applications on the ClassCloud architecture.

Table 2.
Comparison between physical and virtual cluster in the ClassCloud

	Physical cluster	Virtual cluster in ClassCloud
Flexibility	Low	High
Speed of deployment	Slow	Fast
Cost	High	Low
Utilization	Specific	Wide
Performance	High	Low
Management	Easy	Hard

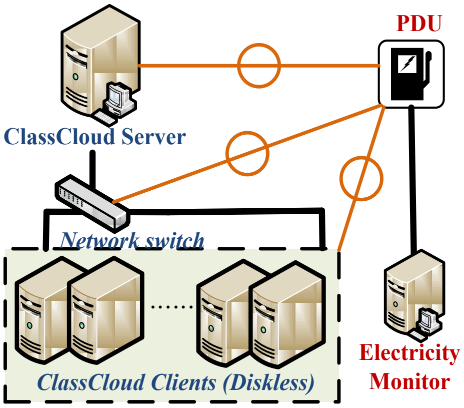

Fig. 4. Power evaluation architecture of ClassCloud

5.2. Power consumption analysis and statistics

In this section, there are three different types for power consumption test. The following test types are: (1) Power consumption of booting stage; (2) Power consumption of I/O-bound test; and (3) Power consumption of CPU-bound test. According to different booting modes in this test, there are three cases which are referred to Case 1: local boot with HD, Case 2: PXE boot with HD, and Case 3: PXE boot without HD. The ClassCloud project takes the Case 3 to build the Cloud Computing platform. In the following sections, we make analysis of the power consumption evaluation with the following formula in next section.

5.2.1 Power consumption calculation

The PDU provides an Ethernet interface for coupling to the electricity monitor and it records the total current of the testbed every minute. User can observe the current variation and download relative data from electricity monitor. In order to automate these three types of the power consumption test, we combine all of the command lines and steps into a shell script and it exports logs automatically with time stamps, including start and end time of each test step. Because of the time limit of time stamps can be precise to the second, the difference between PDU timer and script time stamps is a noticeable problem that we should concern.

Consider the power consumption calculation model in Fig. 5. The two red lines indicate the start and end time of the actual time duration individually. The actual total power usage W, represented in Fig. 5. as the area between test start time (t_0) and end time (t_1) should be calculated from the area between PDU time T_3 and T_0 through the transformation ratio of the time difference. Electric unit of the area between t_0 and t_1 is referred to ampere-hour (Ah), so we can use the equation (1) below to calculate the total power used in unit kilowatt hour.

The total power usage of each test can be calculated by using the following equation:

$$W = \left[\frac{(a+b)}{2} + \frac{(b+c)}{2} + \frac{(c+d)}{2}\right] \times \frac{1}{60} \times \frac{t_1 - t_0}{T_3 - T_0} \times \frac{V_s}{1000}$$
$$= \frac{(a+b)+(b+c)+(c+d)}{120} \times \frac{t_1 - t_0}{T_3 - T_0} \times \frac{V_s}{1000} \quad (1)$$

W:	*total power usage (Kilowatt hour)*
V_s:	*utility power source voltage (Volt)*
a,b,c,d:	*electric current usage (Amp)*
T_0,T_3:	*PDU sample time duration (Minutes)*
t_o,t_1:	*actual time duration (Minutes)*

5.2.2. Power consumption of booting stage

As shown in Fig. 6, the x-axis stands for the cases of different booting modes, and the y-axis is the average kilowatt hour in power consumption. The booting stage is referred to the period from Power-On state to operating system idle state.

It is obviously shown that booting via PXE helps to save more power than using hard disks. The power consumption variation of the network switch and ClassCloud server is nearly neglectable in any working modes in power evaluation experiment, but booting via PXE may sometimes take more time to finish booting. We consider that the reason for taking more time to boot via PXE is that all of these ClassCloud clients are waiting for the IP address assignment from ClassCloud server, and it also takes some times to download necessary booting data, especially when the ClassCloud server is in heavy load. Therefore, it might be an intractable problem in a great quantity of PCs.

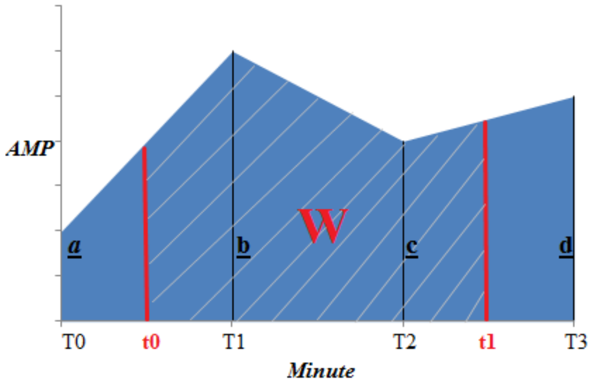

Fig. 5. Power consumption calculation model

5.2.3. Power consumption of I/O-bound Test

In this section, we use two GNU/Linux commands "dd" to create an empty 1GB image file for the first case and "tar" to make a file compression of this 1GB image file created from command "dd" as the second case. In addition, we also have Bonnie++ hard disk performance test for the third case of the I/O-bound test.

The Fig. 7 shows the comparison between three different cases of booting modes by using GNU/Linux command "dd". We can see that both of the PXE boot have worse power saving than local boot because of the high throughput I/O data communication and overhead on the ClassCloud server. Besides, it really takes a long time to create empty images repeatedly for each clients on same hard disk of the server.

As shown in Fig. 8, the comparison between three different cases of booting modes by using GNU/Linux command "tar" is still the same circumstance compared to Fig. 7. The high throughput I/O data communication and overhead are also the main reasons for worse power saving.

The Fig. 9 shows the comparison between three different cases of booting modes by using Bonnie++ hard disk benchmark suite. The circumstance is still the same compared to Fig. 7. Therefore, high I/O intensive applications and tasks are not suitable for the ClassCloud architecture.

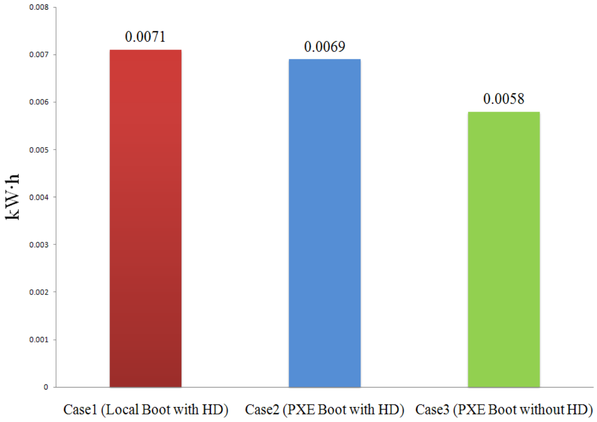

Fig. 6. Power consumption of booting stage

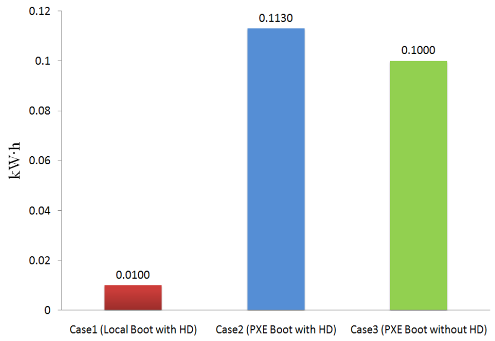

Fig. 7. Power consumption of GNU/Linux "dd" command

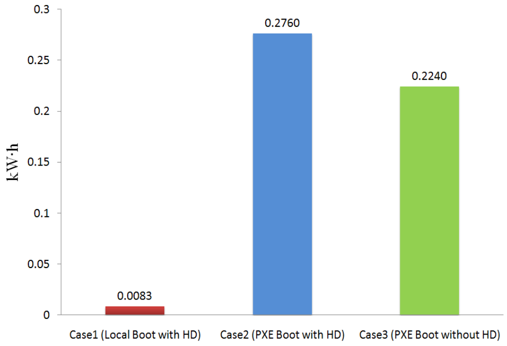

Fig. 8. Power consumption of GNU/Linux "tar" command

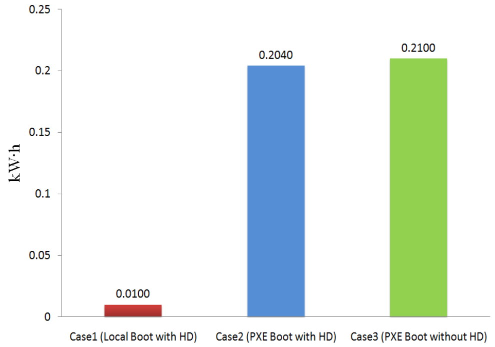

Fig. 9. Power consumption of Bonnie++ benchmark test

Table 3.

Power saving percentage of Case3 in power evaluation experiment

	Case3 vs. Case1[a]	Case3 vs. Case2[b]
BootTime	18.31%	15.94%
dd	-900%	11.50%
tar	-2598.79%	18.84%
Bonnie++	-2000%	-2.94%
Make kernel –j 1	22.22%	10.64%
Make kernel –j 2	33.33%	21.74%
Make kernel –j 3	16.59%	12.76%
Make kernel –j 4	10.67%	7.56%

[a] **Case3 vs. Case1 = (Case1-Case3)/Case1×100%**
[b] **Case3 vs. Case2 = (Case2-Case3)/Case2×100%**

5.2.4. Power consumption of CPU-bound test

The Fig. 10. shows the comparison between three different cases of booting modes in using GNU/Linux command "make" with different cores of CPU. The x-axis stands for the number of CPU cores in both three cases, and the y-axis is the average kilowatt hour in power consumption of different booting modes with different CPU cores.

In order to get rid of the I/O obstruction, we use the RAM disk for temporary storage. It is obviously shown that the power consumption of the Case 3 is always the lowest of the others. Owing to the cache of CPU and RAM disk, it makes the power saving more efficiently and economically. Hence, it is recommended to run high CPU intensive applications on the ClassCloud architecture.

5.2.5. Discussion

By the means of the power evaluation experiment, we can say that the disklesss design of DRBL really brings notable effect on power saving for CPU-bound applications. By acquiring the average kilowatt hour in booting stage, command test and benchmark test of these three cases, it shows that the power consumption of Case 3 in CPU-bound test is always the lowest of the others, and it saves almost 33.33% power compared to Case 1, 21% power compared to Case 2. Table 3 shows the power saving percentage of Case 3 in power evaluation experiment. Owing to the bottleneck of high throughput I/O data communication and overhead, the ClassCloud architecture is not suitable for I/O intensive applications. On the other hand, it brings a great benefit to those CPU intensive applications with RAM disk. Table 4 is the time consuming of these three cases in the experiment, and it provides a relative comparison of the power and performance. It obviously shows that the Case 3 also has better performance than the others in CPU-bound test.

Table 4.

Time consuming of the three cases in power evaluation experiment

	Case1	Case2	Case3
BootTime	14 secs	31 secs	31 secs
dd	29 secs	317 secs	313 secs
tar	23 secs	802 secs	799 secs
Bonnie++	28 secs	605 secs	629 secs
Make kernel –j 1	127 secs	108 secs	108 secs
Make kernel –j 2	64 secs	57 secs	57 secs
Make kernel –j 3	44.5 secs	41 secs	41 secs
Make kernel –j 4	34 secs	35.5 secs	35.5 secs

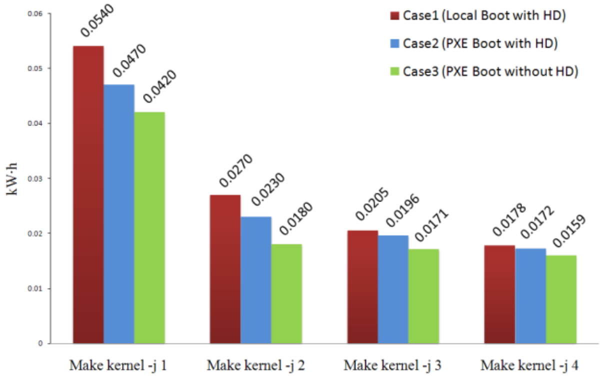

Fig. 10. Power consumption of GNU/Linux "make" command

6. Conclusion

In recent years, the Cloud Computing and Green Computing have become the most popular and well-known buzzword after Web 2.0, and it brings innovative thoughts to IT industries and academia. The purpose of our project is to provide a useful Cloud Computing deployment toolkit to those who want to build up an energy efficient Cloud Computing infrastructure for education or research. In fact, PC classrooms can become a potential computing center in any idle time. That's the reason why we open the project named ClassCloud to help those who want to turn their PC classrooms into a small-scale computing center for Cloud Computing quickly and efficiently. In addition, the power evaluation experiment shows that the DRBL really works on power saving based on the technique of diskless in the ClassCloud project. The discussion of carbon footprint and energy conservation is more and more popular, and we all consider that Green Computing should be recognized as a kind of social responsibility. Those who use information technology to benefit society and better living should to face this issue squarely.

Facing the challenge of Green Computing in the future, we still have lots of performance tuning and power measuring work for advanced evaluation. Some practical tools like bootchart [2], lm-sensors [6] and powertop [10] will be adopted in the following experiment for advanced power evaluation. Besides, some hardware techniques like Ramdisk, solid-state drives and RAID are also to be considered as future work for improving High-throught I/O performance. On the other hand, the DRBL has already provide a convenient management user interface and capability for ClassCloud platform, but the lack of well-managed capability in the ClassCloud virtualization infrastructure is insufficient. For this reason, we have begun to integrate the OpenNebula [8] into ClassCloud for better virtualization infrastructure management, and it will help a lot in hybrid Cloud with different PC classrooms.

References

[1] Amazon.com, Inc., *Amazon Web Services: Overview of Security Processes*. Sep. 2008. [Online]. Available: http://s3.amazonaws.com/aws_blog/AWS_Security_Whitepaper_2008_09.pdf

[2] (2009) The Bootchart website. [Online]. Available: http://www.bootchart.org/

[3] *Green Computing webpage on Computing.co*. [Online]. Available: http://www.computing.co.uk/computing/specials/2162404/green-computing

[4] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," *Grid Computing Environments Workshop, 2008. GCE '08*, vol., no., pp.1-10, 12-16 Nov. 2008.

[5] J. Cope, M. Oberg, H. M. Tufo, and M. Woitaszek, "Shared Parallel Filesystems in Heterogeneous Linux Multi-Cluster Environments," *proceedings of the 6th LCI International Conference on Linux Clusters: The HPC Revolution*, 2005.

[6] (2009) The lm-sensors website. [Online]. Available: http://lm-sensors.org/

[7] *Diskless Remote Boot in Linux (DRBL)*, NCHC. [Online]. Available: http://drbl.sourceforge.net/

[8] The OpenNebula website. [Online]. Available: http://www.opennebula.org/

[9] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the Art of Virtualization," *in SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*. New York, NY, USA: ACM Press, 2003.

[10] (2009) The PowerTop website. [Online]. Available: http://www.lesswatts.org/

[11] R. J. Figueiredo, P. A. Dinda, and J. A. B. Fortes, "A case for grid computing on virtual machines," in *ICDCS '03: Proceedings of the 23rd International Conference on Distributed Computing Systems*. Washington, DC, USA: IEEE Computer Society, 2003.

[12] Green Computing definition webpage on SearchDataCenter.com. [Online]. Available: http://searchdatacenter.techtarget.com/sDefinition/0,,sid80_gci1246959,00.html

[13] W. C. Kuo, C. Y. Tu, Y. T. Wang, "Deploy Kerrighed SSI Massively Using DRBL," *HPC ASIA 2009*, 2009.

[14] *Green Computing*, Wikipedia. [Online]. Available: http://www.greenlivingpedia.org/Green_computing.

[15] W. Huang, J. Liu, B. Abali, and D. K. Panda, "A case for high performance computing with virtual machines," in *ICS '06: Proceedings of the 20th annual international conference on Supercomputing*. New York, NY, USA: ACM Press, 2006, pp. 125-134.

[16] (2009) The Xen website. [Online]. Available: http://www.xen.org/