

# 國立交通大學

## 資訊科學與工程研究所

### 碩士論文

大學程式能力檢定系統設計、實作與部署：

一個全虛擬化的實例

The Design, Implementation, and Deployment of the CPE  
(Collegiate Programming Examination) System –  
an Application of Full Virtualization

研究生：陳奕任

指導教授：黃世昆 博士

中華民國 一〇一 年 五 月

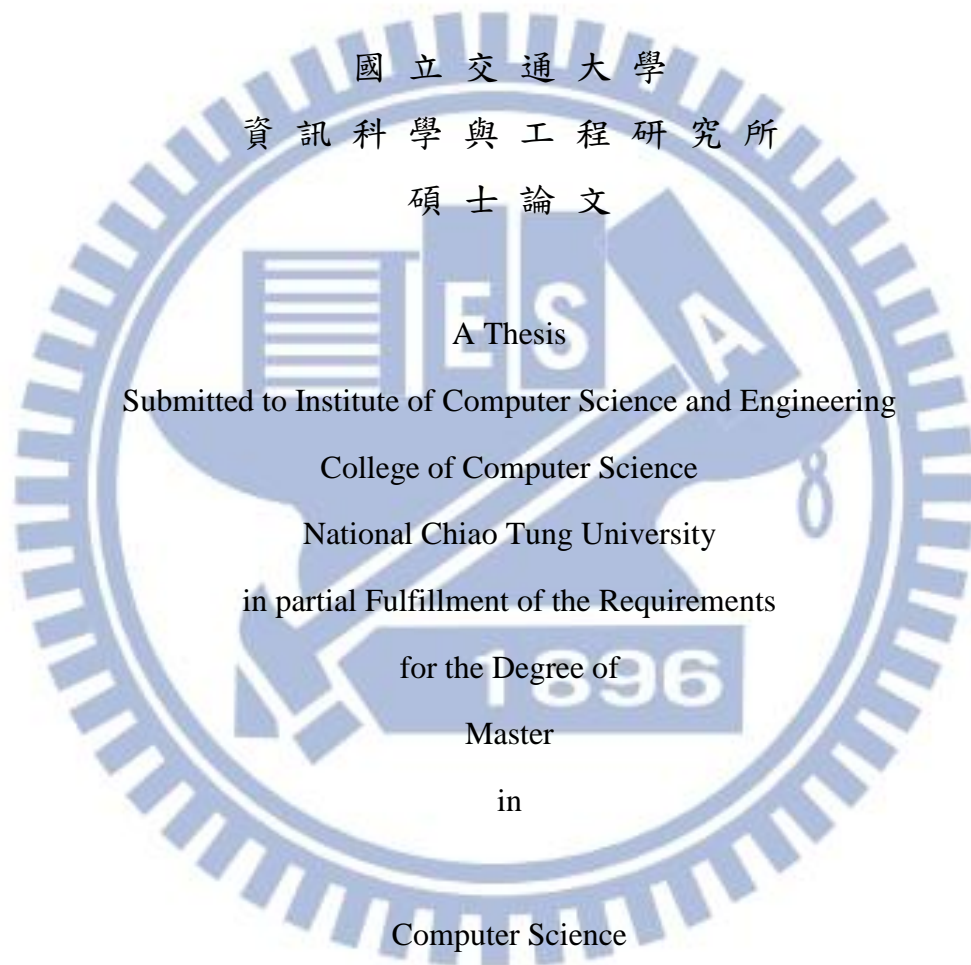
大學程式能力檢定系統設計、實作與部署：一個全虛擬化的實例  
The Design, Implementation, and Deployment of the CPE (Collegiate  
Programming Examination) System – an Application of Full Virtualization

研 究 生：陳奕任

Student : Yi-Ren Chen

指導教授：黃世昆 博士

Advisor : Dr. Shih-Kun Huang



May 2012

HsinChu, Taiwan, Republic of China

中華民國 一〇一 年 五 月

# 大學程式能力檢定系統設計、實作與部署：一個全虛擬化的實例

學生：陳奕任

指導教授：黃世昆 博士

國立交通大學資訊科學與工程研究所碩士班

## 摘 要

大學程式能力檢定始於中華民國九十九年，最初由國立交通大學和國立中山大學聯合舉辦。在試驗眾多軟體與方法後，本論文藉由撰寫程式以整合多樣化的技術，使得這樣大型的跨校聯合上機考試成為可能。我們提出了能夠方便又快速部署考場環境的解決方案以減輕電腦教室管理人員的負擔。因此在 2012 年五月舉行的一次大學程式能力檢定已有 33 所國內大學參與。我們發展的這套系統能夠管理 1700 台的虛擬機器，可以做為雲端計算服務的管理方式。

# The Design, Implementation, and Deployment of the CPE (Collegiate Programming Examination) System - an Application of Full Virtualization

Student: Yi-Ren Chen

Advisor: Dr. Shih-Kun Huang

Institute of Computer Science and Engineering  
National Chiao Tung University

## Abstract

Collegiate Programming Examination(CPE) has been launched in 2010, and initially organized by National Chiao Tung University and National Sun Yat-sen University. This work includes implements to integrate a variety of technologies in the trial of a number of software, and contributed to such a large-scale inter-collegial joint examination of computer operation. We propose a convenient and rapid solution for deployment of examination classroom environment to reduce the burden of administrators of the computer classrooms. There are 33 universities joining CPE held in May 2012. We have developed the mechanism in order to manage thousands of virtual machine. And this mechanism could be contributed to the management of cloud computing service.



# Acknowledgements

First of all, I would like to express my sincere gratitude to my advisor, Prof. Shih-Kun Huang, for his guidance and patience. Without his encouragement, I would not complete this thesis. Thank also to all the members in Software Quality Laboratory for their reinforcement and suggestion. I especially appreciate Yong-Ren Yang, who has been worked in the National Chiao Tung University Information Technology Service Center several years. He has tested the system repeatedly and maintained the cluster of Collegiate Programming Examination. He is the most conscientious employee I have seen. Thank also to Prof. Chang-Biao Yang at the University of Sun Yat-Sen, for his passional promotion for Collegiate Programming Examination, and his important contribution to the computer science on higher education in Taiwan. Moreover, I want to thank to all the people who ever supported me during these days. Finally, I will sincerely dedicate this thesis to my parents and my favorite angel, Wei-Ni Chu.

# Contents

摘 要 .....	i
Abstract.....	ii
Acknowledgements.....	iii
Contents .....	iv
List of Tables .....	vi
List of Figures.....	vii
<b>Chapter 1 Introduction.....</b>	<b>1</b>
1.1 Motivation.....	2
1.2 Objective .....	4
<b>Chapter 2 Related Work.....</b>	<b>7</b>
2.1 ACM ICPC World Finals Contest Image.....	7
2.2 Solutions for replication.....	8
2.3 The Technology of Virtualization.....	8
2.4 Cloud Computing .....	9
<b>Chapter 3 Method .....</b>	<b>11</b>
3.1 Decide To Use VirtualBox.....	12
3.2 Decide To Use FreeBSD .....	12
3.3 Reducing Size of Virtual Disks.....	13
3.4 Restricting the Examinees' Privileges to Use FreeBSD .....	14
3.5 Preventing Examinees from escaping from The Virtual Machine.....	15
3.6 Restoring a Clean Examination Environment Rapidly .....	16
3.7 Provide A Variety Way Of Installations .....	17

3.8	Simplify The Setup Procedure By Single Installation Executable.....	18
3.9	Assign An Unique Identifier To Every Virtual Machine.....	18
3.10	Backup Examinees' Source Code Automatically .....	19
3.11	Flexible Expansion Of The Central Control Architecture .....	20
<b>Chapter 4</b>	<b>Implementation .....</b>	<b>21</b>
4.1	Installing FreeBSD in VirtualBox.....	21
4.2	Reducing the Size of Virtual Disks.....	22
4.3	Escape Prevention .....	24
4.4	The Single Highly Integrated Installation Executable .....	25
4.5	Non-tracking Peer-to-peer Replication over The Local Area Network.....	26
4.6	The Management Agent.....	27
4.7	The Central Control Architecture.....	27
<b>Chapter 5</b>	<b>Results and Evaluations.....</b>	<b>30</b>
5.1	System installation .....	30
5.2	File Replication .....	31
5.3	Stress Testing .....	32
<b>Chapter 6</b>	<b>Conclusion.....</b>	<b>34</b>
<b>Reference.....</b>		<b>35</b>

# List of Tables

Table 1 Comparison of The Varied Implementations..... 11

Table 2 Particular Layout of Virtual Disks ..... 23

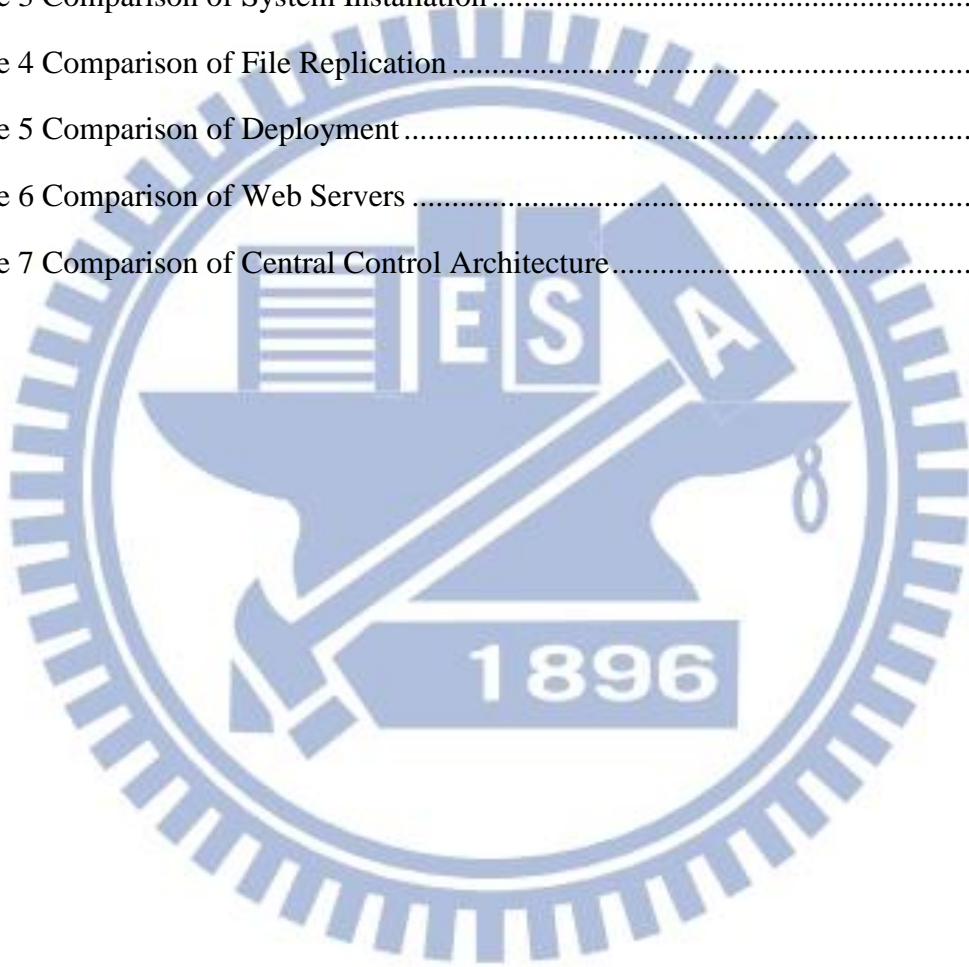
Table 3 Comparison of System Installation ..... 30

Table 4 Comparison of File Replication ..... 31

Table 5 Comparison of Deployment ..... 32

Table 6 Comparison of Web Servers ..... 33

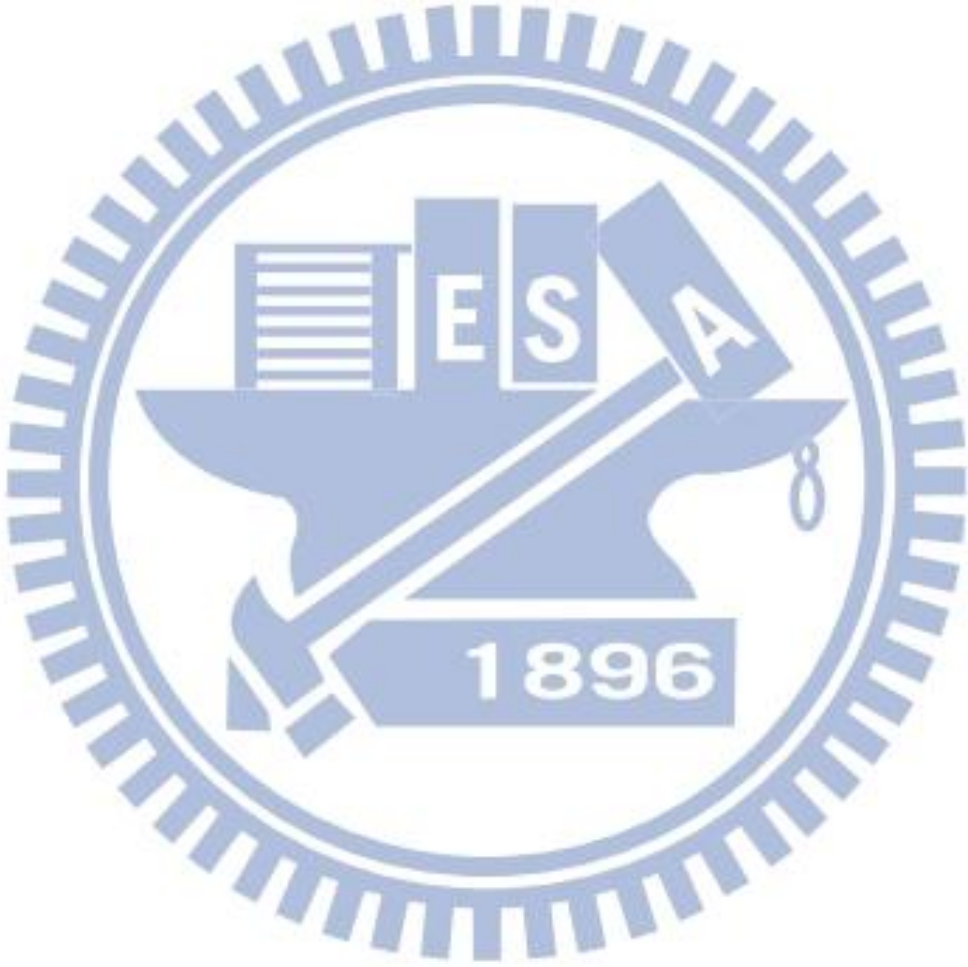
Table 7 Comparison of Central Control Architecture..... 33





# List of Figures

Figure 1 Central Control Architecture .....28



# Chapter 1 Introduction

Beginning in 2008, the Department of Computer Science in National Chiao Tung University launched a proficiency examination of programming. Students must solve several programming problems in the restricted hardware and software environment, upload their source code through the network to the web-based judge system, and then the results of judge will be displayed on the web page. Initially, the ways to hold the examination in the Department of Computer Science are creating an unprivileged account for the students logging on Windows, and setting the firewall rules on routers in order to restrict them to only connect to the judge system.

There are several computer classrooms in the Department of Computer Science. The computers in these classrooms are connected with the same local area network, and obtain an IP address via DHCP. The external network connections are unrestricted in the case of general use, but during the examination, switching the configuration of routers is necessary. We may not book the same classroom for every examination, and this will cause trouble in the switching the configuration of routers. Sometimes, it would cause inconvenience for users because of network outages of all the computer classrooms.

Adding or removing software in most of computer classrooms is frequent because of the demand for temporary needs of courses. After a period of time, software installed on computers in the same classroom is almost similar except slight differences, but it is discouraged in the situation of examinations. Some administrators use Reborn card with hard disk partition in order to switch between the examination and general purpose. It seems suitable for administrators if all requests of vary courses and examinations were already known.

The computer center in the Department of Computer Science at National Chiao Tung University has managed computers without Reborn card or diskless system for several years. This is reasonable because we have to upgrade one of computer classrooms in one or two years averagely. Reborn card and diskless systems are limited to the consistency of the hardware environment, and installing corresponding drivers in the operating system is necessary. During the transition from 32-bit to 64-bit operating systems, many manufacturers have not developed the 64-bit version of the driver, so the rapid deployment of consistent software environment become a challenge.

## 1.1 Motivation

There have been many famous online judge systems since 1995. One of them is

"<http://uva.onlinejudge.org/>". Problem set collected in their database and presented in the form of website. Registered members can try to solve any of these problems and submit their source code. The system will automatically schedules to judge the results of submission and sends notification to them via e-mail. This site also retains the record of each member's problem-solving. It is the shared memories of many excellent software engineers.

Another software tool related to programming contests is "PC<sup>2</sup>". It provides teams to submit their source code, in addition to automatic scheduling to judge the results of submission, it updates the scoreboard immediately. But we have to install the client program on every computer in the classroom, remove unnecessary software on these computers, and switch the configuration of routers, in order to hold an examination.

The proficiency examination of programming launched by our department needs the online judge system hosted in our department. The system retains the record of each student's problem-solving as their learning profile and plus with feature of real-time scoreboard. This system has been powered by "DomJudge". Although "DomJudge" has mixed the advantages of online judge system and on-site contest system, the main trouble still is to rapidly switch the software environment on computers and

configuration of routers between the examination and general purpose.

In domestic universities, many computer science-related institutes and departments hope to enhance the programming proficiency of the students through examinations.

They also often held examinations in the respective unit. Therefore, they are facing the similar issues to us, even if they able to use Reborn card or diskless system. Some of them planned to hold a joint examination, but never realized. In addition to a lot of trouble in the switch between the examination and general purpose, communication and coordination of technical and administrative level are very difficult.

Our goal is to find a solution to simplify the complexity of holding an examination in the computer classroom. And making the dreams of holding joint examinations come true.

## 1.2 Objective

Our ultimate goal is to deploy the environment for programming proficiency examination in thousands of personal computer scattered in computer classrooms in domestic universities. In order to achieve this goal, we have proposed an approach of deployment which has simple, fast and stable features. Enough to convince the heavy



workload administrators of computer classroom are willing to take a little time to setup this environment.

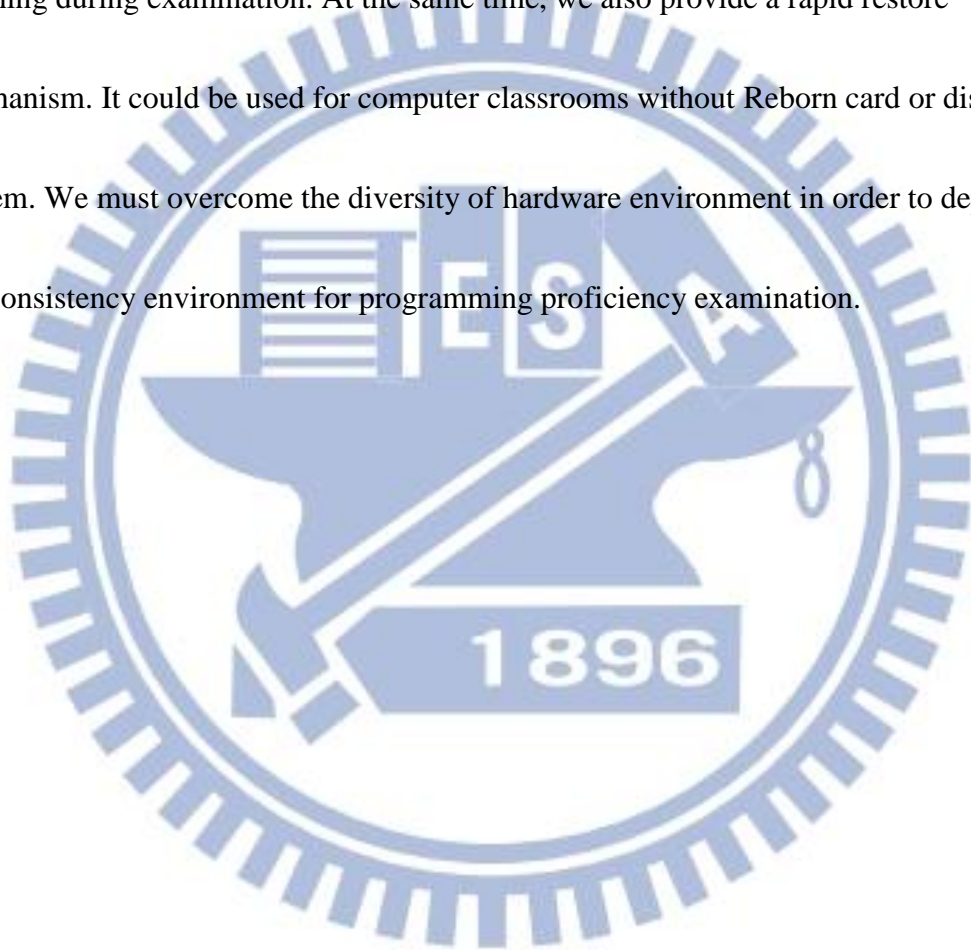
In the past few years, it requires a lot of preparatory work to hold a programming contest. For fairness, a certain number of computers with the same specification are required. For most of computer classrooms, it is not a problem. With Reborn card, we might need a new partition on the hard disk to install a clean operating system. With diskless system, we might clone a clean disk image and install necessary software. For administrators, to install a clean operating system and necessary software is a time-consuming work. Using Reborn card or diskless system to deploy the environment also takes a moment depending on the size of the disk partition and network bandwidth.

We will provide an installation program for administrators to install the environment for programming proficiency examination, and ensure the environment is clean. In general case, administrators will not need to adjust configuration of routers, because we have configured the security settings to avoid cheating through network in the examination. The environment is not only easy to install but also easy to completely uninstall, so administrators are able to rapidly switch computer classrooms between examination and general usage.

To rapidly distribute files over the local area network, we also propose a

peer-to-peer approach without centralized tracking. It is suitable for the computer classroom without Reborn card or diskless system. It is very lightweight, and even without installing any software.

Stability was also seriously considered. We will test and tune repeatedly, and avoid crashing during examination. At the same time, we also provide a rapid restore mechanism. It could be used for computer classrooms without Reborn card or diskless system. We must overcome the diversity of hardware environment in order to deploy the consistency environment for programming proficiency examination.



## Chapter 2 Related Work

### 2.1 ACM ICPC World Finals Contest Image

ACM ICPC World Finals Contest Image [5] is available at "<http://pc2.ecs.baylor.edu/InstallDirections.html>". Until 2012, the newer one is available at "<http://pc2.ecs.baylor.edu/ImageBuildInstructions.2012.html>". By comparing the old and new contest images, we found that there are several changes: They replaced CentOS with Ubuntu, and used the full installation disc rather than a simple boot disc. These changes may be due to that Ubuntu is more suitable for desktop environment than CentOS and the full installation disc simplifies the installation.

Packaging the contest environment into an installation disc is a good idea. This makes the administrators of the contest site do not need one by one to install the particular software specified by the contest. The installation disc is bundled with the contest control system, so the administrators are able to setup the contest environment easily and need not worry about the security issues during the contest.

## 2.2 Solutions for replication

Reborn card is a hardware solution for replication. Firstly, it uses the technique of hardware interrupt to protect a partition. It combines the functionality of the network adapter afterward, and has the ability to replicate disk partitions over the network. It is due the increase in the bandwidth of Ethernet, in recent years, the emergence of a technology is called "diskless system". It combines a variety of specification developed, such as PXE booting and iSCSI. Since disk images are managed on the server, if the computers in the local area network are booting almost the same time, the delay is obvious.

Norton Ghost has long been famous among the backup software. It can back up the entire disk or just a partition into an image file. Ghost supports multicast since version 4.0, so it can be used for efficient replication in a computer classroom. There is also an open source clone system called "Clonezilla". The target of Clonezilla is intended to supersede Norton Ghost.

## 2.3 The Technology of Virtualization



The technology of virtual machine has been mature for years. However, because the arrival of the multi-core era, it began to be popular. Virtualization provides many advantages, such as sharing resources, migration across platforms, and reducing down time. Virtual machines usually use a file as a virtual hard disk. This is much like the concept of disk image used by Ghost and Clonezilla, but the purpose is different. Ghost and Clonezilla must write the disk image into a real disk or partition, and then the computer is ready to boot. Virtual machines can directly boot from their virtual hard disk. Innate limitations of the virtual machines are that it is not real, so its performance has not been able to be compared with the real machine.

## 2.4 Cloud Computing

The client-server model is used in the mainframe era, so cloud computing is seems like the mainframe. Cloud computing is also a kind of distributed and parallel computing. And it combined with virtualization technology, enabling flexible adjustment of resource usage.

Cloud computing usually provides three different types of service: Software as a Service(SaaS), Platform as a Service(PaaS), and Infrastructure as a Service(IaaS). IaaS



offers computers whatever physical or virtual ones, so it needs physical data center management (PDCM) and virtual data center management (VDCM). The computer in the cloud is usually called "node", and identifying the nodes in the cloud is a non-trivial issue. PaaS offers the development and production environment, such as web servers, databases, development tools, and function libraries. SaaS offers application software for consumers.

The definition of SaaS includes several services such as remote desktop. In this paper, we also provide a fast way to construct a desktop environment. It can be considered as a kind of SaaS. PaaS is to offer a platform for development and production. Many providers focus on the platform of web service, such as Google App Engine. In this paper, we provide a convenient development environment for C++ and Java programmers. It can be considered as a kind of PaaS. In order to identify and manage hundreds of distributed virtual machines, we have implemented a part of VDCM which is the base of IaaS.

# Chapter 3 Method

In order to build a consistent of the examination software environment, the following solutions have been evaluated:

1. Boot from the USB flash
2. Boot from the optical disc
3. Remote desktop connection
4. Virtual machine

Table 1 Comparison of The Varied Implementations

<b>Implementations</b>	<b>Advantages</b>	<b>Disadvantages</b>
Boot from the USB flash	<ol style="list-style-type: none"> <li>1. Does not require pre-installed.</li> <li>2. Based on hardware performance.</li> </ol>	<ol style="list-style-type: none"> <li>1. Slightly more difficult to create image.</li> <li>2. High purchase cost.</li> <li>3. Security issue.</li> </ol>
Boot from the optical disc	<ol style="list-style-type: none"> <li>1. Does not require pre-installed.</li> <li>2. Based on hardware performance.</li> </ol>	<ol style="list-style-type: none"> <li>1. Slightly more difficult to create image.</li> <li>2. High depletion rate.</li> <li>3. Security issue.</li> </ol>
Remote desktop connection	<ol style="list-style-type: none"> <li>1. Does not require pre-installed.</li> </ol>	<ol style="list-style-type: none"> <li>1. Heavy load of the remote server.</li> <li>2. Need a stable network connection quality.</li> <li>3. High construction cost.</li> </ol>
Virtual machine	<ol style="list-style-type: none"> <li>1. Low cost</li> <li>2. Stable operation</li> </ol>	<ol style="list-style-type: none"> <li>1. Required pre-installed.</li> <li>2. Lower performance.</li> </ol>

### 3.1 Decide To Use VirtualBox

After evaluations, we believe that the most feasible solution is using virtual machine. At first, we would like to try to use the VMware. Because in our experience, its performance is better than the products of other virtual machines. However, it is a commercial product, and cannot be distributed without agreement. So we finally decided to use VirtualBox which is open source software.

### 3.2 Decide To Use FreeBSD

And then I follow the recommendations and accordance with the directions on this web page "<http://pc2.ecs.baylor.edu/InstallDirections.html>", to install ACM ICPC World Final Contest image in VirtualBox. Because this image is specifically designed for the ACM ICPC World Final Contest, many of the operating system management tools have been removed. It is very difficult to add or remove software to meet our needs, so we began to imitate the contest image, to install a new operating system and the software needed by us in the virtual machine.

We have tried to install CentOS just like the contest image. Due to the complex dependency of Linux packages, we could not easily remove unnecessary packages. If

we re-built package from source code, it may reduce the complex dependency, but much like using FreeBSD. Linux kernel supports more modern hardware devices; however, at this time, we may not need these drivers if the faster boot time is seriously considered.

As a server operating system, FreeBSD is fairly famous. Few people would use FreeBSD as a desktop environment. X window is not the default option during FreeBSD installation, and to configure X window to run on FreeBSD may encounter a variety of difficult problems. I had experience of using X window on FreeBSD a few months, So I knew that X window could be ran on FreeBSD efficiently and stably.

### 3.3 Reducing Size of Virtual Disks

Virtual disks are usually stored in the form of files. While creating a virtual disk, there are usually two options: "allocate all space at once" or "dynamically grow when need". In order to save network transmission time, we cannot allow the virtual disk to be allocated all space at once. However, the dynamic growth of the virtual disk has a characteristic that is never be reduced after growth. If we would like to minimize the size of the virtual disks, we must not install unnecessary packages on FreeBSD.



However, this idea is impractical because there are always some of the packages used to assist in the installation of other packages, but they are never needed at run time. So, installation of packages for programming proficiency examination is needed to split to two stages. At the first stage, we perform the normal installation of packages on FreeBSD and remove the unnecessary packages at the end of this stage. And then we create several new virtual disks and copy files from the original virtual disks to these new virtual disks at the second stage. Such a complex procedure is to ensure that the virtual disks only grow to the appropriate size.

### 3.4 Restricting the Examinees' Privileges to Use FreeBSD

As a desktop environment used for the examination, examinees do not need to have root privileges on FreeBSD. In order to exempt the invigilators' trouble, the system will automatically login as an ordinary user, and examinees will be blocked from trying to login as root. Restricted external network communication become essential to examinations fairness. To conduct such examinations in the past, usually taken to unplug the external network wire or modify the firewall rules of network devices and



recover the network connection after the end of the examination, this practice may cause trouble for administrators. So we have configured firewall rules on the virtual machine in advance, and the firewall rules can be modified by the management agent during the examination. The management agent will periodically access to our central control server via HTTP. Administrators usually do not block the HTTP connection, unless they execute a proxy policy. Our server is accessed in passive mode to avoid Network Address Translation (NAT) issue.

### 3.5 Preventing Examinees from escaping from The Virtual Machine

Most of virtual machine products with graphical user interface can run in the full screen mode. They can return to window mode by pressing specific key combinations. But we do not permit examinees to manipulate the virtual machine to run in window mode, so we must disable these specific key combinations to make the virtual machine operate in kiosk mode. We must also disable the toolbar of virtual machine, but doing so is not enough. Microsoft Windows itself has a key combination that will not be intercepted by any application, that is, "Ctrl-Alt-Delete". After pressing the key

combination, the task manager will be invoked. Users could terminate any process by using the task manager if they have sufficient privileges. Users may use the task manager to switch applications to foreground, so we have to make every effort to block invoking the task manager.

Although we have just prevented examinees from escaping from the virtual machine, no one can guarantee that the examinees would obediently start the virtual machine and switch it to the full screen mode. So I wrote some programs to automatically start the virtual machine and make it run in the full screen mode. Therefore, examinees are enforced to stay inside the virtual machine.

### 3.6 Restoring a Clean Examination Environment Rapidly

In the beginning of every examination, we have to clean the user's home directory. That is also one of the reasons why administrators of the computer classroom are unwilling to hold the examination. Administrators need to manually one by one clean up, and nobody could substitute or help them due to privilege issues. Now, a clean home directory can be restored after the virtual machine boot. Our management agent

fetches the clean home directory which is compressed in the form of file from the central control server. It could be done quickly because the size of the compressed file is quite small. And it does not ask administrators to do anything additionally.

We also provide a recovery mechanism. By simply click your mouse twice, and then the virtual machine will be repaired in few minutes, because we have locally retained an original archive of the virtual machine. This mechanism is especially suitable used for the computer classroom without Reborn card or diskless system.

### 3.7 Provide A Variety Way Of Installations

We have defined a modular directory structure. The structure has been created at our FTP site. Administrators can download it to any disk device. And we export the virtual machine in the form of open virtualization format archive with the file extension "ova". Administrators can put an "ova" file into the corresponding directory in order to install our system.

We have proposed an approach for rapidly replication over the local area network. It inherits the peer-to-peer concept but without tracking, and therefore is decentralized.

In order to ease administrators' overhead, we implemented this approach by writing

batch file and copying files via Common Internet File System (CIFS). This approach is especially suitable for the computer classroom without Reborn card or diskless system.

### 3.8 Simplify The Setup Procedure By Single Installation Executable

It seems trivial to deploy the unified software environment to lots of computer by using virtual machine. But some administrators will resist doing this if the manual procedure takes efforts in more than one minute. The normal installation of products of virtual machine has some unconfirmed options. And we also need to do additional settings to prevent cheating in the examination. These operations are definitely exceeding the administrators' tolerable limit. This is why the large-scale joint examination of the use of computers has never been held in Taiwan. So a single highly integrated installation executable can ease the work of the system administrators.

### 3.9 Assign An Unique Identifier To Every Virtual Machine



To distinguish from these duplicated virtual machines, we make the management agent to perform the registration on the central control server. The identifier of a virtual machine will be dynamically changed every time after it boots. We must not assign the static identifier to a virtual machine, because we distribute it from a single source to lots of destination. We also must not locally store the identifier assigned by the server, because the previously registered virtual machine may be replicated by the various way. The management agent has the ability to detect whether the examinee has logged in judge system. And then reports the unique identifier of the virtual machine used by the examinee to the judge system.

### 3.10 Backup Examinees' Source Code Automatically

Many programmers may have the same experience that sometimes the crashes happen in debugging their program. It is easy to forget to backup manually, so regularly automatic backup may be helpful, but the broken source code would be submitted through the regularly automatic backup mechanism. Many of software engineers are accustomed to commit their code into the version control system. Most of version control system is not easy to use and the careless engineers may make the mistakes to

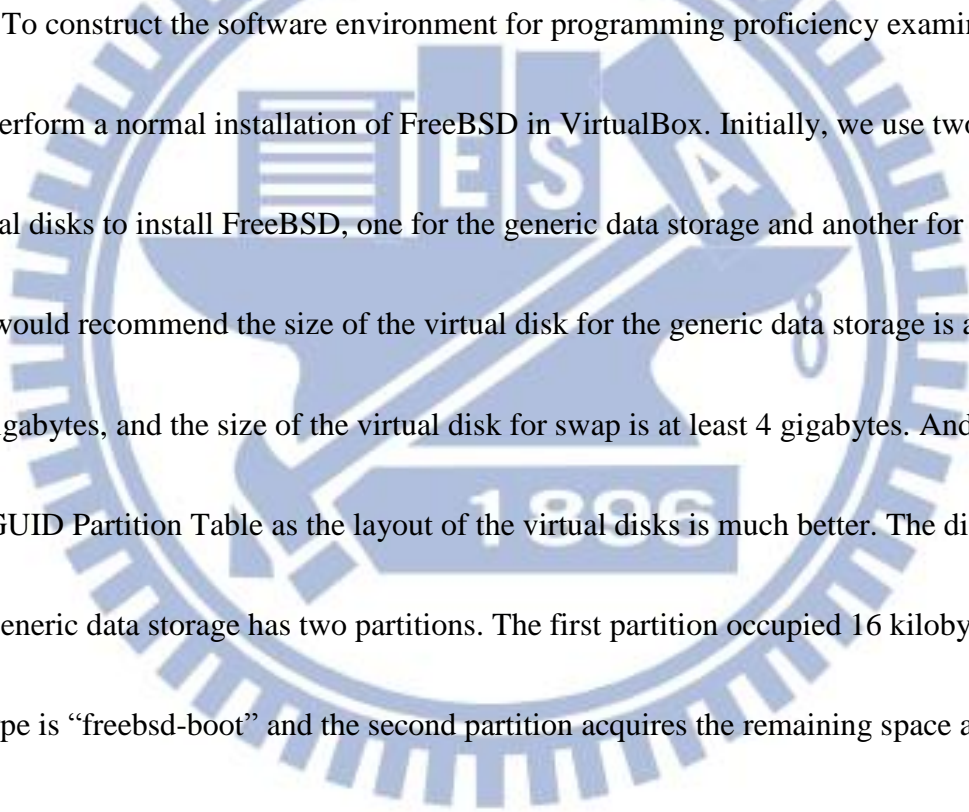
operate. Without any configuration, our backup mechanism would be performed at the beginning of compiling source code. And it performs a differential backup because many programmers often just do a little modification of their source code while debugging.

### 3.11 Flexible Expansion Of The Central Control Architecture

Our central control server has a cluster of servers to support it. While the HTTP request is reached the central control server, it uses port forwarding to the rear cluster by using round robin in order to balance the loading. Each server in the cluster runs the two services: hypertext and databases. Since databases are distributed, it is easy to add the additional servers into the cluster.

# Chapter 4 Implementation

## 4.1 Installing FreeBSD in VirtualBox



To construct the software environment for programming proficiency examination, we perform a normal installation of FreeBSD in VirtualBox. Initially, we use two virtual disks to install FreeBSD, one for the generic data storage and another for swap. We would recommend the size of the virtual disk for the generic data storage is at least 16 gigabytes, and the size of the virtual disk for swap is at least 4 gigabytes. And using the GUID Partition Table as the layout of the virtual disks is much better. The disk for the generic data storage has two partitions. The first partition occupied 16 kilobytes and its type is “freebsd-boot” and the second partition acquires the remaining space and its type is “freebsd-ufs”. The other disk for swap has only one partition occupying the whole space of the disk and its type is “freebsd-swap”.

We install software by using ports instead of pre-build packages to avoid incompatibility. To reduce the disk usage, we customized the ports of Xorg and GNOME by modifying their Makefile. And we installed the “virtualbox-ose-additions”

rather than any other video driver for Xorg to automatically adjust the display resolution in full-screen mode.

“Code::Blocks” is an integrated development environment for C and C++ programming language. It uses “xterm” as its console for debugging, but we would like to use “gnome-terminal” instead of “xterm”. So we patched “Code::Blocks” for this purpose.

“Eclipse” is an integrated development environment for Java programming language. It could also be used for C and C++ programming language if the plug-in called "CDT" was installed. Because the version of CDT in the ports is too old to be compatible with Eclipse in the ports, we manually installed CDT by using the plug-in management tool in Eclipse. The CDT plug-in requires a newer version of GNU debugger than the built-in, so we also installed the suitable version of GNU debugger.

After all required software installed; we rebuild and install the kernel. And then we remove some unnecessary packages. Finally, we do a detail configuration for installed software and set the preference for the examinees.

## 4.2 Reducing the Size of Virtual Disks



Although the virtual machine is ready to distribute for programming proficiency examination, the size of the virtual disks should be inflated. It would be unfavorable to distribute because some colleges have the limit of downloading traffic per day.

To minimize the size of virtual disks, we do the following steps: First, we copy the disk for the generic data storage to a new one named “duplicated system” and add it to the virtual machine. And then add 7 new virtual disks to the virtual machine.

Table 2 Particular Layout of Virtual Disks

File Name(Max. size)	Label	Size of partition	Comment
Maintained-system.vdi (512MB)	/	128MB	
	/usr	(remaining space)	
Maintained-swap.vdi (4GB)	(none)	4GB	
Maintained-vartmp.vdi (8GB)	/tmp	1GB	
	/var	(remaining space)	
Maintained-local.vdi (3GB)	/var/db/pkg	24MB	newfs -i 8192 -b 16384 -f 2048
	/usr/local/etc	48MB	
	/compat	224MB	
	/usr/local	(remaining space)	
Maintained-home.vdi (16GB)	/root	1GB	
	/home	(remaining space)	
Maintained-ports.vdi (12GB)	/var/db/ports	8MB	newfs -i 8192 -b 16384 -f 2048
	/var/db/portsnap	256MB	newfs -i 8192 -b 16384 -f 2048
	/usr/ports	(remaining space)	
Maintained-srcobj.vdi	/usr/src	1GB	

(3GB)	/usr/obj	(remaining space)	
-------	----------	-------------------	--

Second, on the disk named “duplicated system”, we modified the configure files in order to enhance security and cleaned unnecessary files in these directories and their sub directories: “/tmp”, “/var”, and the examinee’s home directory.

Third, within the “duplicated system” disk, we compressed the above-listed directories in reverse order. And then extracted these compressed files to the corresponding partitions on the new virtual disks.

Finally, we removed these seven new virtual disks on the virtual machine and add the first five of them to a new virtual machine. Before not yet booting the new virtual machine, we exported it to the open virtualization format archive.

### 4.3 Escape Prevention 1896

To prevent examinees' escaping from virtual machine; after the examinee logged on Windows, we will run the virtual machine in the full-screen mode. Moreover, the toolbar and host key of the virtual machine are disabled, and the task manager on Windows is also disabled.

By referencing VirtualBox User Manual [7] and MSDN library [9], these configurations seem not difficult for the experienced administrators, but with a little

detailed steps. So we have integrated those configurations into a single installation executable.

## 4.4 The Single Highly Integrated Installation

### Executable

In addition to prevent from escaping, the single installation executable was integrated with installation of GNU wget, downloading of the specified VirtualBox's version, reinstallation of VirtualBox, account creation on Windows and importing the virtual machine for the new created account.

In our previous trial, the user profile was not found at the ending of account creation but it would be initialized at the first time of logging on Windows. We have tried to fake a similar user profile directory after creating an account programmatically, but that did not totally succeed in different versions of Windows. So, in the past, administrators needed to manually create a new account, to log as the newly created account and then to import the virtual machine if they would like to install our system.

We improve the process in the following. The Windows command “runas” is much like the UNIX command “sudo”, and it could be executed with an argument “/profile”.

It can not only simulate the acts of importing the virtual machine as an unprivileged user but also initialize the user profile for newly created account. Eventually, we integrated all our works into a single self-extractable executable.

## 4.5 Non-tracking Peer-to-peer Replication over The Local Area Network

For the consistency of software environment in the computer classroom, to replicate files over the local area network may be a regular work. We have developed the utility to speed up for this purpose. Additionally, the utility was implemented by using only batch file on Windows. We utilize the built-in file-sharing mechanism by invoking the "net" command.

The utility has two components: one for the server, and another for the client. While the component of server is running, it starts to share the directory with allowing only one client. While the component of client is running, it starts to infinitely search available server in order to copy the shared files and it will sleep a few seconds if it reaches an invalid server. After finish of copying, the client will get the component of server, and then run it. So the number of server would be exponentially increasing and



the total remaining time of replication would be logarithmically decreasing.

## 4.6 The Management Agent

The management agent is a simple script invoked at the boot time of the virtual machine. It will fetch a plain text file as successive script from our central control server and then invoke the successor. This mechanism is very powerful but also very dangerous. We need neither to replace nor to modify thousands of distributed virtual machine. You can instantly change the directive that they will perform. So it is not hard to imagine why we could accomplish those features, and we can easily append more features in the future.

## 4.7 The Central Control Architecture

The current architecture has ten servers: one is named “balancer”, another is named “backend”, and the others are called “attendants”. The balancer runs Domain Name Server (DNS) and is configured as Network Address Translation (NAT) router. The backend runs Network File System (NFS) server and Network Information Service (NIS)

server for all the other servers. The attendants run Apache Tomcat web server and Oracle MySQL database server. These Tomcat servers are configured as a cluster and use multicast to replicate session data. The MySQL servers can run independently.

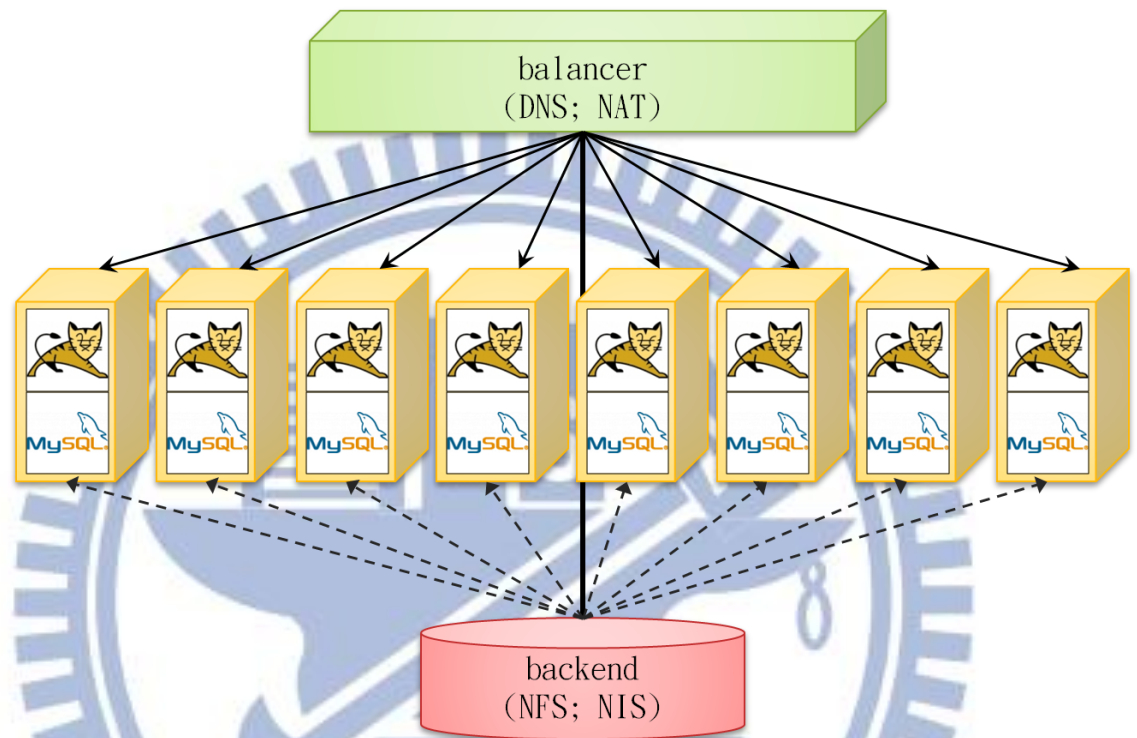
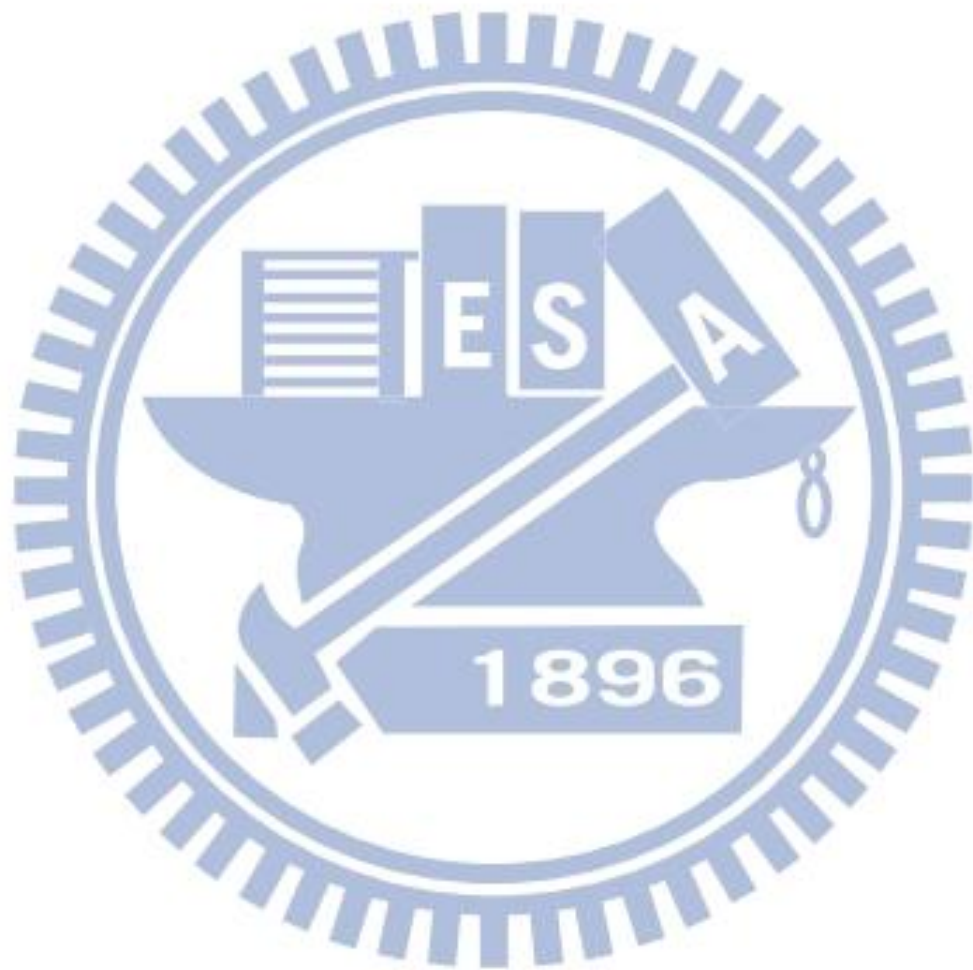


Figure 1 Central Control Architecture

For incoming HTTP requests, the balancer does a round robin port forwarding to one of attendants. If a request causes a new session creation, the appointed attendant handled this request by itself and save its hostname to the newly created session. So for a request with old session, the appointed attendant acted as a proxy to forwarding request to the last attendant in the session. Each Tomcat running on the server only connects to the MySQL database server running on the same host.

Databases in these MySQL servers have the same schema, but tables in the

database are independent. Because we stored a table into distributed database servers, the table in one of the database server was shred and it could be easily increased in the number of attendants.



# Chapter 5 Results and Evaluations

## 5.1 System installation

The comparison of system installation among ACM-ICPC World Final Contest and Collegiate Programming Examination is listed below:

Table 3 Comparison of System Installation

	ACM-ICPC World Final Contest	Collegiate Programming Examination
Form of deployment	Linux installation disc image	Open Virtualization Format Archive
Size of downloaded file	About 700 MB	About 1 GB
Time of installation	About 15 minutes	About 5 minutes
Time of operation during installation	About 5 minutes	About 30 seconds
Allowed root login	Yes	No
Required to set IP address	Yes	No
Required to configure firewall	Yes	No
Support recovery	No	Yes
To clean user home directory	Manual operation	By remote control
Multi sites supporting	No	Yes



Our system installation has many benefits than theirs. So our approach is highly adaptable.

## 5.2 File Replication

We have proposed the non-tracking peer-to-peer replication approach over the local area network and it has been verified as our expectation. There was an experiment that files which are totally about 1 GB in size were be copied to 50 computers via 100 Mbps LAN. The time spent in this experiment is about 9 minutes. The number of servers could exponentially increase and the total remaining time of replication could logarithmically decrease.

Table 4 Comparison of File Replication

Number of Destination	SMB/FTP	Non-tracking Peer-to-peer
1	1 unit of time	1 unit of time
3	3 unit of time	2 unit of time
7	7 unit of time	3 unit of time
15	15 unit of time	4 unit of time
31	31 unit of time	5 unit of time
63	63 unit of time	6 unit of time

Table 5 Comparison of Deployment

	Diskless System	Reborn Card/ CloneZilla	Non-tracking Peer-to-peer
Cost	High	Medium/ Low	None
Installation time	About 5 minutes	About 5 minutes	About 25-35 minutes ( $5 + 0.5 * N$ )
Replication time	About 5 minutes (boot concurrently)	About 60 minutes (depend on disk usage)	About 8 minutes ( $1.333 * 6$ )
Summation	About 10 minutes	About 65 minutes	About 33-43 minutes

Our approach has reasonable cost-effective than others.

### 5.3 Stress Testing

As a web server, Apache is fairly famous. But it must be configured by experienced administrators to cope with the heavy load. In the earlier implementation, Apache forks a new process to handle the incoming request. Although it has been implemented the threaded workers, in our previous trials, Apache could only handle hundreds of concurrent request.

In contrast, Tomcat is based on Java technology with natively threaded feature.

Undoubtedly, it is more adaptable to the growing service.

Table 6 Comparison of Web Servers

	Apache	Tomcat
Memory usage per request	About 2-4 MB	About 300-600 KB
Memory manager	Operating System	Java Virtual Machine
Dynamic page technology	PHP (interpret)	JSP (compile once)
Connection pool for database	Manually use “pconnect” function	Automatically managed by JDBC

Table 7 Comparison of Central Control Architecture

	previous	current
Type of web server	Apache (Dynamic) + Lighttpd (Static)	Tomcat
Number of web server	1 + 1	8
Expandable	Hard	Easy
Load Balance	Yes	No
maximum transient requests	About 900-1200	About 8000-16000

## Chapter 6 Conclusion

Now, the programming proficiency examination is weekly held in the Department of Computer Science at National Chiao Tung University. The examination would not cause toil to administrators of the computer center. Since the system is convenient and rapid to install, many colleges are willing to join. Eventually, we have succeeded in regularly holding large-scale inter-collegial joint programming examinations. This system is continuous improvement in all aspects of installation and execution efficiency. It has experienced real stress test to reach stability. Today, we are pleasure to announce that we have achieved our goals. Our central control architecture has the ability to manage thousands of virtual machine and it may be used for Virtual Data Center Management (VDCM). Moreover, we can provide a programming platform in a few minutes. This is a kind of platform as a service (PaaS) and an example of cloud computing.



# Reference

- [1] Dohan Kim, "Flexible virtual machines in distributed systems and grids", School of Computer Science, University of Windsor
- [2] Joseph Clifton, "A Simple Judging System for the ACM Programming Contest", Computer Science and Software Engineering University of Wisconsin
- [3] Arefin, A., et al., "Secured Programming Contest System with Online and Real-time Judgment Capability", International Conference on Computer and Information Technology, Bangladesh, 2005
- [4] Tongrang Fan, Zhanwei Liu, Liping Niu, "A Study Of Dragon-lab Federal Experiment Cloud and Network Contest", School of Information Science and Technology, Shijiazhuang Tiedao University, China
- [5] ACM ICPC World Finals Contest Image Installation Directions,  
<http://pc2.ecs.baylor.edu/InstallDirections.html>
- [6] FreeBSD Handbook,  
[http://www.freebsd.org/doc/en\\_US.ISO8859-1/books/handbook/](http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/)
- [7] VirtualBox User Manual, <https://www.virtualbox.org/manual/UserManual.html>
- [8] HTTP/RFC2616, <http://www.w3.org/Protocols/rfc2616/rfc2616.html>

[9] MSDN library, <http://msdn.microsoft.com/en-us/library/>

[10] Apache Tomcat 7 Docuemnt,

<http://tomcat.apache.org/tomcat-7.0-doc/index.html>

