# *A Tutorial on Setting up Ubuntu Linux Virtual Machines*

**Copyright 2009**

**Dr. Lixin Tao**
**http://csis.pace.edu/lixin**
**Pace University**

PDF Version: http://csis.pace.edu/lixin/ubuntu/LinuxSetup.pdf
Video Version: http://csis.pace.edu/lixin/ubuntu

# Table of Contents

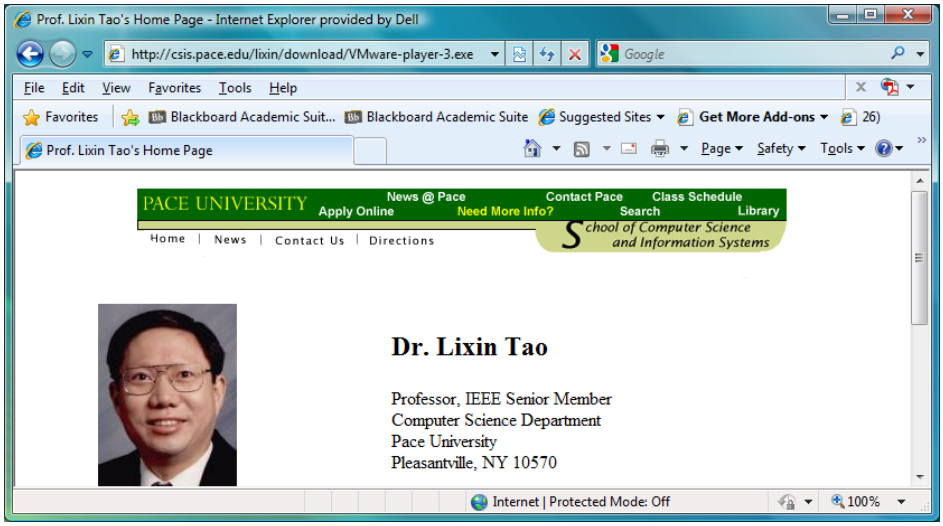

1 Installing *VMware Player* ..... 1
2 Deploying and Using a Completed *Ubuntu* Virtual Machine ..... 5
2.1 Overview ..... 5
2.2 Downloading the Completed Version of *Ubuntu* V9.10 Linux VM ..... 5
2.3 Launching and Logging on the VM ..... 7
2.4 Changing Mouse Focus and Exchanging Files between VM and PC ..... 13
2.4.1 Dragging-and-Dropping ..... 13
2.4.2 Using a USB Portable Disk. ..... 13
2.4.3 Shared Folders ..... 14
2.4.4 Coping-and Pasting Text between VM and PC ..... 16
2.5 Using Basic *Ubuntu* Tools ..... 16
2.5.1 Launching a Terminal Window ..... 16
2.5.2 Launching a Firefox Web Browser ..... 17
2.5.3 Launching a Nautilus File Browser ..... 17
2.5.4 Launching a Text Editor ..... 19
2.5.5 Using Synaptic Package Manager ..... 19
2.5.6 Customizing Ubuntu Application Menu ..... 20
2.6 Finding Out Your Gateway/Router, PC and VM's IP Addresses ..... 21
2.7 Changing Network Connection Method ..... 23
2.8 Running a Cluster of VMs on a Single Computer ..... 24
2.9 Learning Basic Linux Commands ..... 25
2.10 Technical Summary of *Ubuntu* V9.10 and Installed Applications ..... 25
2.10.1 Major Installed Web Architecture Components ..... 25
2.10.2 Installation Credentials and Technical Summary ..... 26
3 Setting Up a Basic *Ubuntu* Virtual Machine ..... 29
3.1 Downloading *Ubuntu* ..... 29
3.2 Creating a *Ubuntu* Virtual Machine ..... 30
3.3 Creating Shared Folder C ..... 40
4 Installing Applications on a Basic *Ubuntu* Virtual Machine ..... 46
4.1 Overview ..... 46
4.2 Downloading Basic Version of *Ubuntu* V9.10 Linux VM ..... 47
4.3 Launching the VM ..... 47
4.4 Adding Menu Item "Open In Terminal" to *Nautilus* File Browser Popup Menu ..... 49
4.5 Installing *Emacs* Text Editor ..... 50
4.6 Running Command *sudo* without Needing Password ..... 52
4.7 Installing 7z ..... 53
4.8 Installing Java JDK, JRE and Java Plugin ..... 53
4.8.1 Downloading Java JDK ..... 53
4.8.2 Installing Java JDK ..... 56
4.8.3 Installing JRE Plugin for Firefox ..... 58
4.9 Installing *Apache Tomcat* ..... 58
4.9.1 Downloading Tomcat Installer ..... 58
4.9.2 Installing Tomcat ..... 59
4.9.3 Tomcat File Organization ..... 61
4.9.4 Changing Tomcat Default Port 8080 ..... 61
4.9.5 Starting Tomcat at Linux Boot-Up Time ..... 62

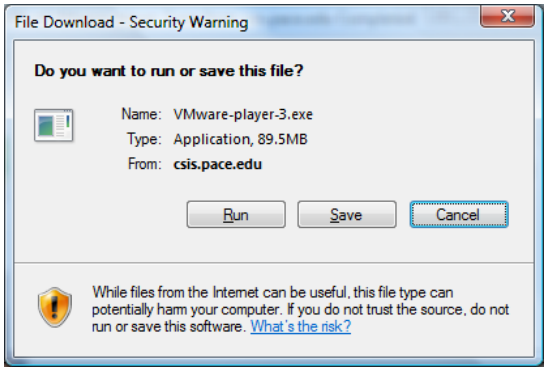
©Copyright 2009 Prof. Lixin Tao
4.10 Installing *Apache* Web Server and Its CGI Plugins ..... 63
4.10.1 Installing Apache 2 ..... 63
4.10.2 Installing PHP ..... 64
4.10.3 Installing Perl ..... 65
4.10.4 Integrating Apache with Tomcat ..... 67
4.11 Installing *MySQL* Database Server ..... 68
4.12 Installing *NetBeans and Sun's Java EE Bundle* ..... 70
4.12.1 Installing Java EE Bundle ..... 70
4.12.2 Developing a Test Java Program ..... 76
4.12.3 Starting/Stopping GlasshFish Application Server and Derby Database Server ..... 80
4.12.4 Adding Tomcat Server to NetBeans ..... 80
4.13 Installing *Eclipse* ..... 83
4.13.1 Downloading Eclipse Installer ..... 83
4.13.2 Installing Eclipse ..... 85
4.13.3 Developing a Sample Java Program ..... 85
4.13.4 Adding Tomcat Server to Eclipse ..... 92
4.13.5 Adding GlassFish Server to Eclipse ..... 95
4.14 Installing Java Tutorials ..... 103
4.14.1 Downloading Sun Java Tutorials ..... 103
4.14.2 Setting Up Java Tutorials ..... 110
4.15 Deploying Example Web Applications on Apache ..... 116
4.15.1 Installing Drupal ..... 116
4.15.2 Installing Sample Tomcat Web Applications ..... 125
5 Linux Basics ..... 130
5.1 Basic Concepts ..... 130
5.2 Linux File System ..... 130
5.3 Linux Accessibility ..... 130

# 1
# Installing *VMware Player*

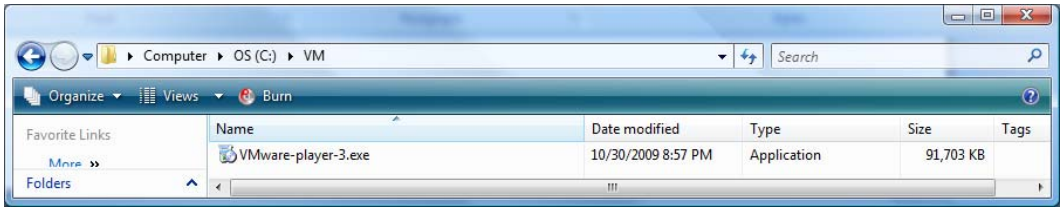
*VMware Player* is a free utility for running VMware Virtual Machines (VMs). It can run one VM a time. If you have *VMware Workstation* on a PC or *VMware Fusion* on a Mac, you can also create VMs and run multiple VMs at the same time. There are *VMware Player* versions for both Linux and Windows, and you can download them directly from http://www.vmware.com/download/player/. Here I explain how to set up *VMware Player* on a Windows PC using a copy of *VMware Player* I prepared for you. You should not install *VMware Player* if you already have *VMware Workstation* or *VMware Fusion* installed on your computer.

- If you have installed *VMware Player* versions earlier than V3.0, uninstall it and reboot your PC.
- In your PC, create a folder "C:\VM" with *Windows Explorer*.
- Use a web browser to visit http://csis.pace.edu/lixin/download/VMware-player-3.exe. Save file "VMware-player-3.exe" in folder "C:\VM" of your PC.

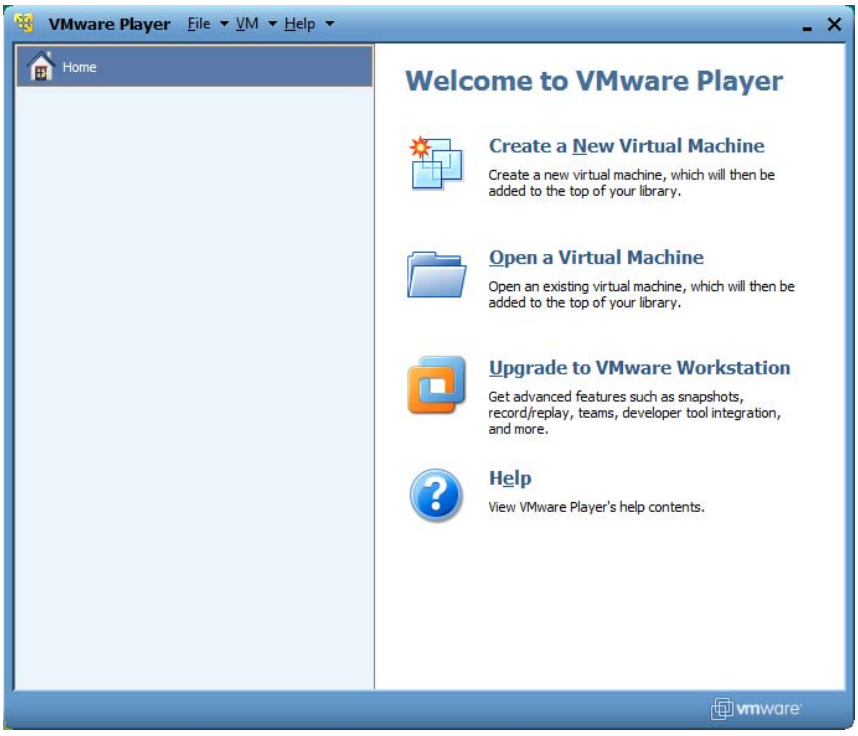


©Copyright 2009 Prof. Lixin Tao



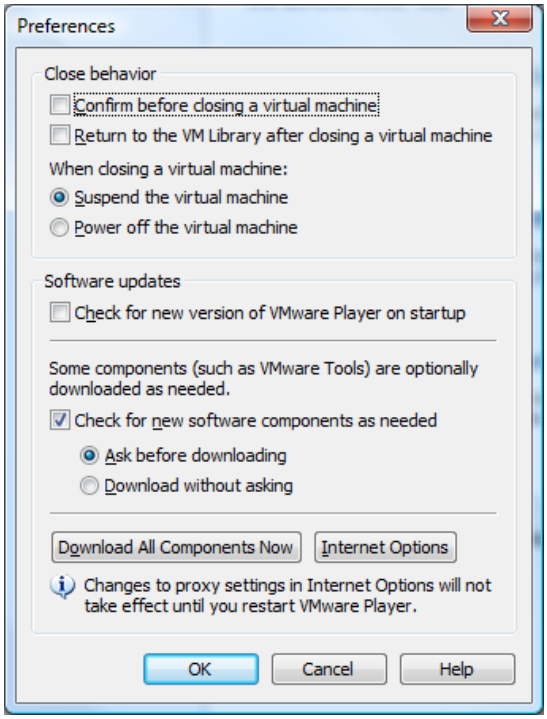

- Double click on file "VMware-player-3.exe" in *Windows Explorer* to install *VMware Player* with default values.
- Reboot your PC.
- Start *VMware Player*, and you will see a window like the following.



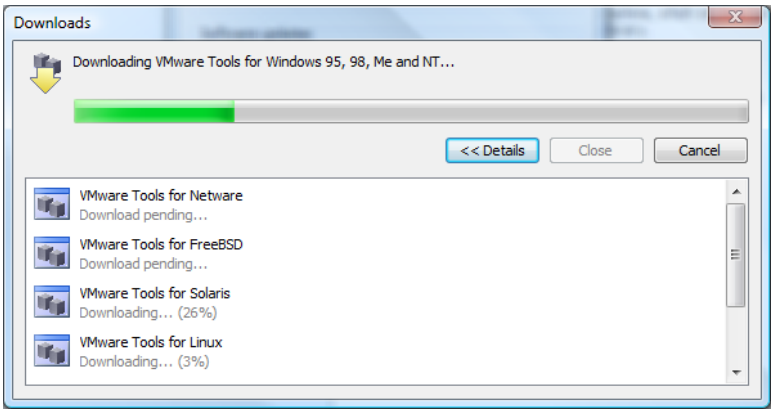

- Click on menu item "File|Preferences…"

3

- In the “Preferences” window, uncheck for software updates, and click on the “Download All Components Now” button so you can later install VMware Tools in your new VMs without Internet access. This step is optional.



©Copyright 2009 Prof. Lixin Tao




- Click on the OK button to close the “Preferences” window.
- Click on the “File|Exit” menu item to exit the “VMware Player” application.

If you are using this tutorial to learn how to install *Ubuntu* and applications on a *Ubuntu* VM, then *VMware Player* is the only software that you need to install on your PC. Your software installation on the VM will have no impact on your PC’s work environment.

# 2
# Deploying and Using a Completed *Ubuntu* Virtual Machine

## 2.1 Overview

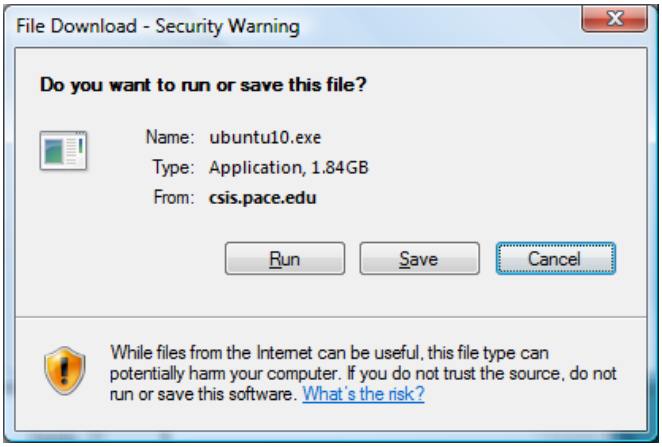
This section guides you to download and install a fully-loaded *Ubuntu* V9.10 VMware virtual machine (VM) named *ubuntu10*. This fully-loaded VM is a good tool for you to learn Linux and web technologies. In the following sections you will learn how to set up a basic *Ubuntu* V9.10 VM named *ubuntu10basic*, and install applications on *ubuntu10basic* to reproduce *ubuntu10* yourself.

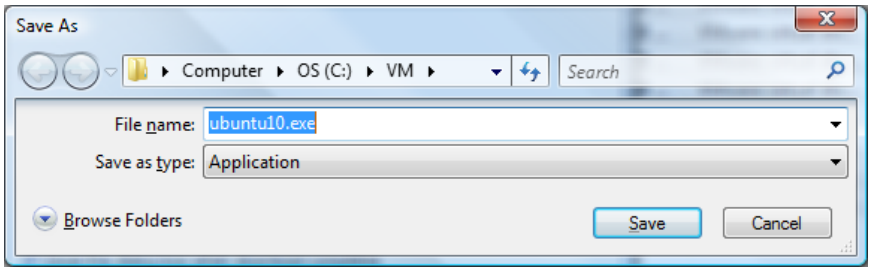
Before you can run a VM, you must have installed the latest *VMware Player* or its equivalent (*VMware Workstation* or *VMware Fusion*). If you have not installed *VMware Player*, refer to the last section to install it first.

## 2.2 Downloading the Completed Version of *Ubuntu* V9.10 Linux VM

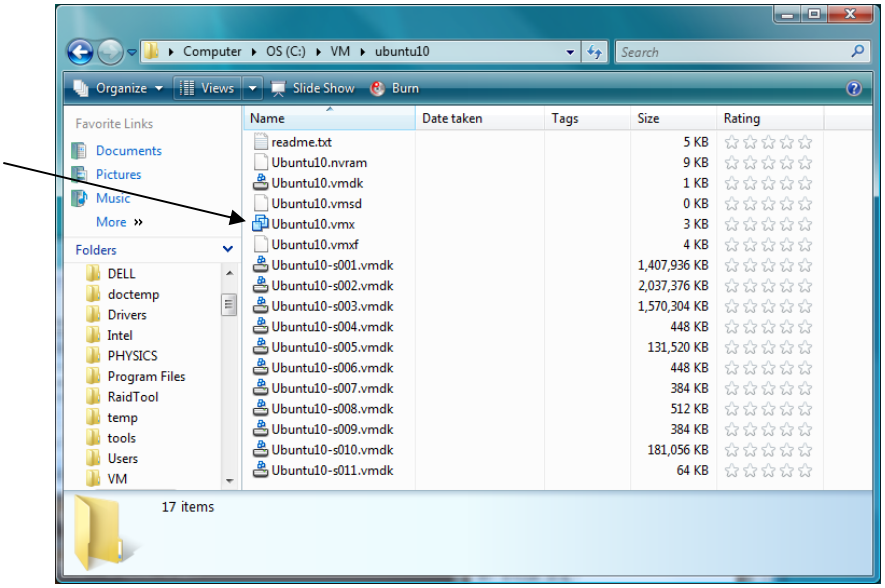
- In your PC, create a folder "C:\VM" with *Windows Explorer*.
- Use a web browser to visit http://csis.pace.edu/lixin/ubuntu/ubuntu10.exe, and save the downloaded file in "C:\VM".

©Copyright 2009 Prof. Lixin Tao




- In a *Windows Explorer*, double-click on file "C:\VM\ubuntu10.exe" to run it, and the execution will generate a folder "C:\VM\ubuntu10".



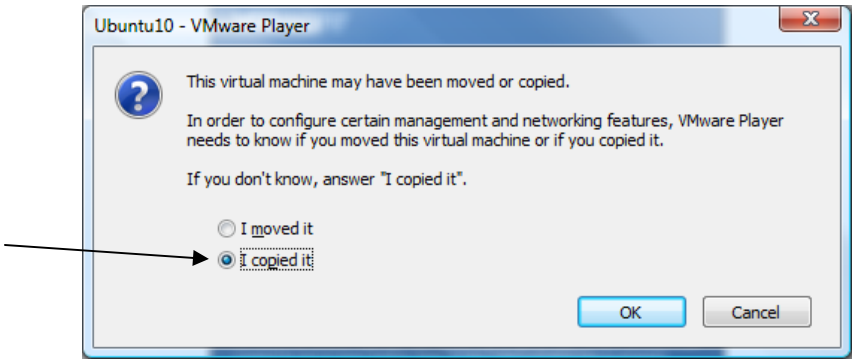

- Save file "C:\VM\ubuntu10.exe" for later regenerating the VM folder if the VM gets compromised.

## 2.3 Launching and Logging on the VM

In folder "C:\VM\ubuntu10", file "readme.txt" tells you some information about this VM including the user names and passwords for this VM. There are two pre-set users "root" and "user", both having password "12345678". To try out the features of this VM, you must logon as "user".

By this time you should have installed the latest version of *VMware Player* (or *VMware Workstation* on a PC, or *VMware Fusion* on a *Mac*), as described in the previous section. Otherwise you would not be able to see the same icons of the files. To launch the VM with *VMware Player*, double-click on file "Ubuntu10.vmx" (if you could not see the file name extension ".vmx", then look for the icon of three partially overlapping blue squares). This file is the configuration file of the VM. Since it is a text file, you could open it with a text editor and make some simple changes, like increasing the memory size for the VM (only if you have more than one GB of physical memory).

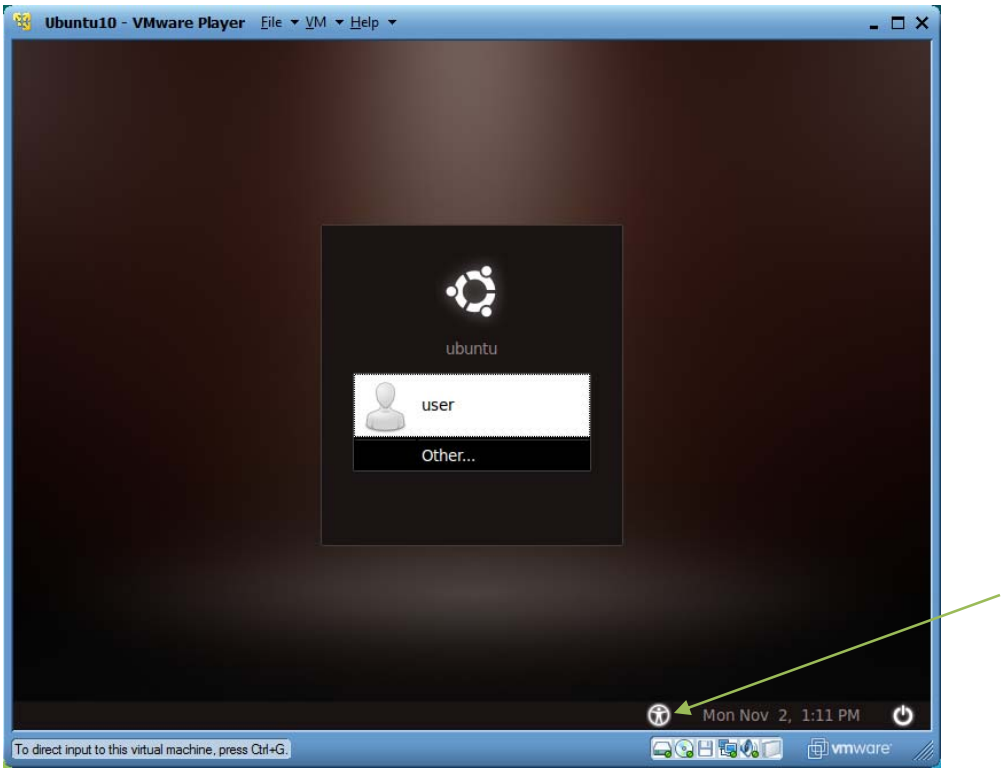
If this is your first time to launch the VM, you may see the following screen:



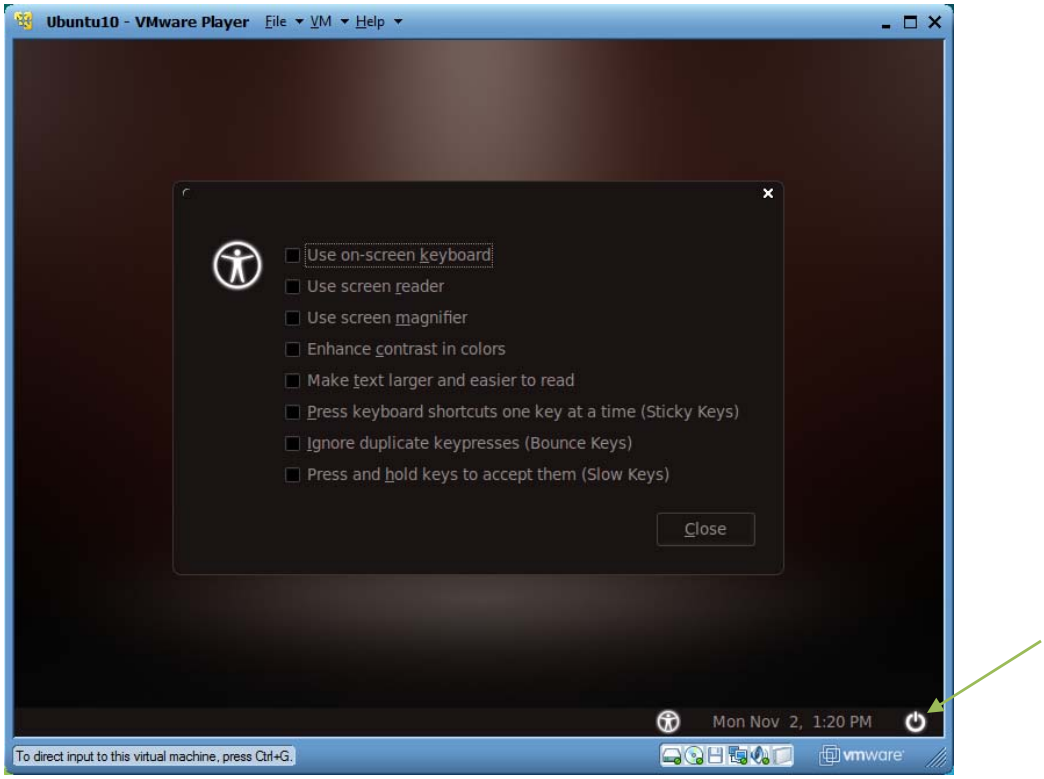

It is better to check the checkbox for "I copied it" and then click on the *OK* button. If you check for "I moved it", the virtual hard disk would keep its unique virtual disk signature used for the licensing of some software as well as its unique virtual MAC (Media Access Control) address of your virtual network card. If you check "I copied it", the virtual hard disk would have a new unique virtual hard disk signature thus may invalidate some software licenses. Since we only install open-source software, they don't make differences for us. If you plan to run multiple VMs concurrently, then each of the VMs must have a unique virtual MAC address so it could get a unique IP address from your DHCP (Dynamic Host Configuration Protocol) server.

After a few seconds you will see the following *Ubuntu* window ready for use.

©Copyright 2009 Prof. Lixin Tao

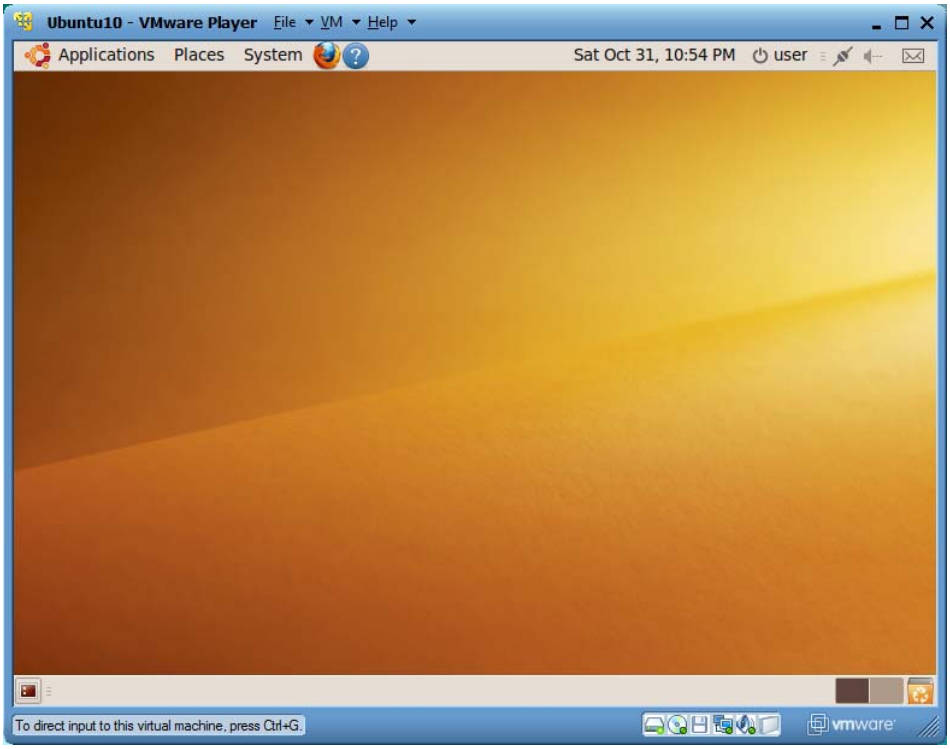



If you click on the person icon in the right-bottom corner, you could set up some user interface details with the following popup window.



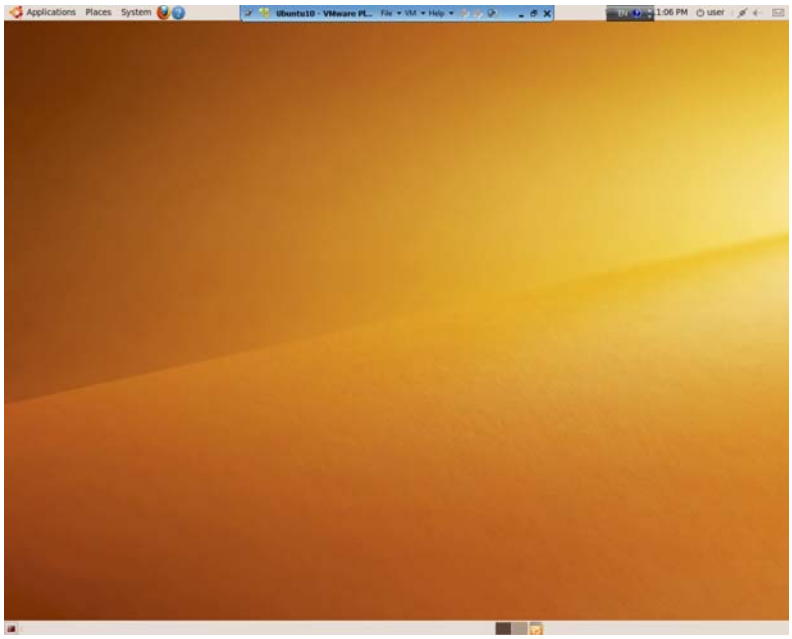
If you click on the power button in the right-bottom corner, then you would have options to hibernate, restart or shut down the VM.

Click on user name “user” (coincident), and enter 12345678 as password, you will see the following screen. If you have created other user accounts on this VM, then you could click on “Other” and type other user names for login.

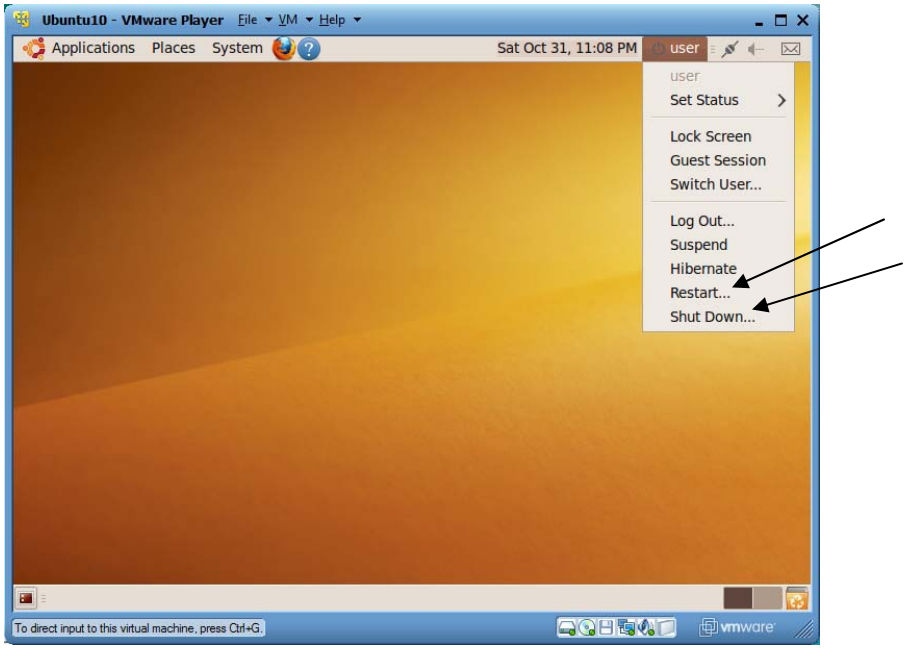



If you maximize the *VMware Player* window, the VM will take over the complete display. If you put your mouse cursor close to the top middle margin, the *VMware Player* menu bar will show up for you to use, as shown below.

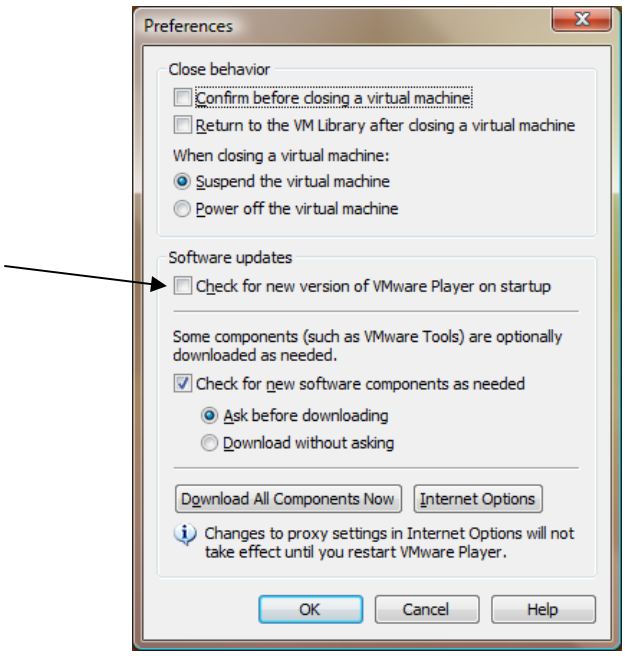
©Copyright 2009 Prof. Lixin Tao



To log out, restart, or shut down the VM, you can click on the right-upper corner "user" (the current user's login name, which happens to be "user"), and then you will be presented with the following choices.

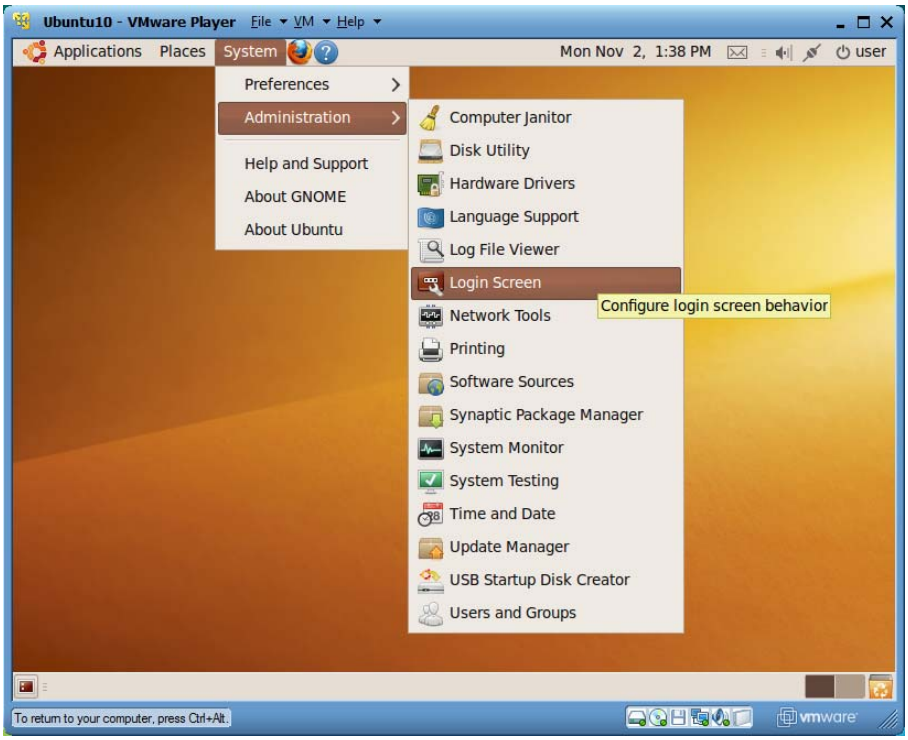



To avoid repeated requests of updating your *VMware Player* during your VM launching, you could click on the top menu item "File|Preferences…", and uncheck the checkbox for "Check for new version of VMware Player on startup", as shown in the next "Preferences" window:




With this same "Preferences" window you can also set close behavior (when you click on the VM's close icon, should the VM suspend or power off the VM, and whether you need to provide a confirmation), and whether *VMware Player* should download all available optional components now or on demand.

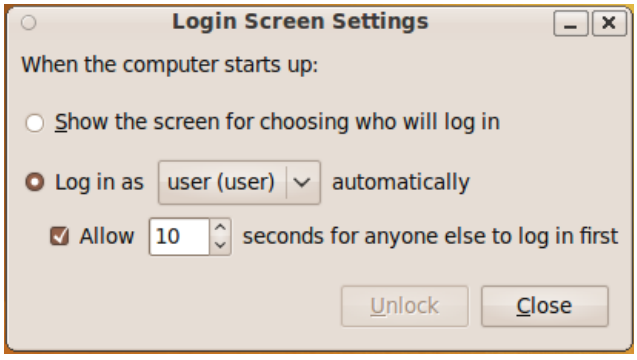
If you prefer the VM to log in as "user" automatically at VM launch time, click on menu item "System|Administration|Login Screen"



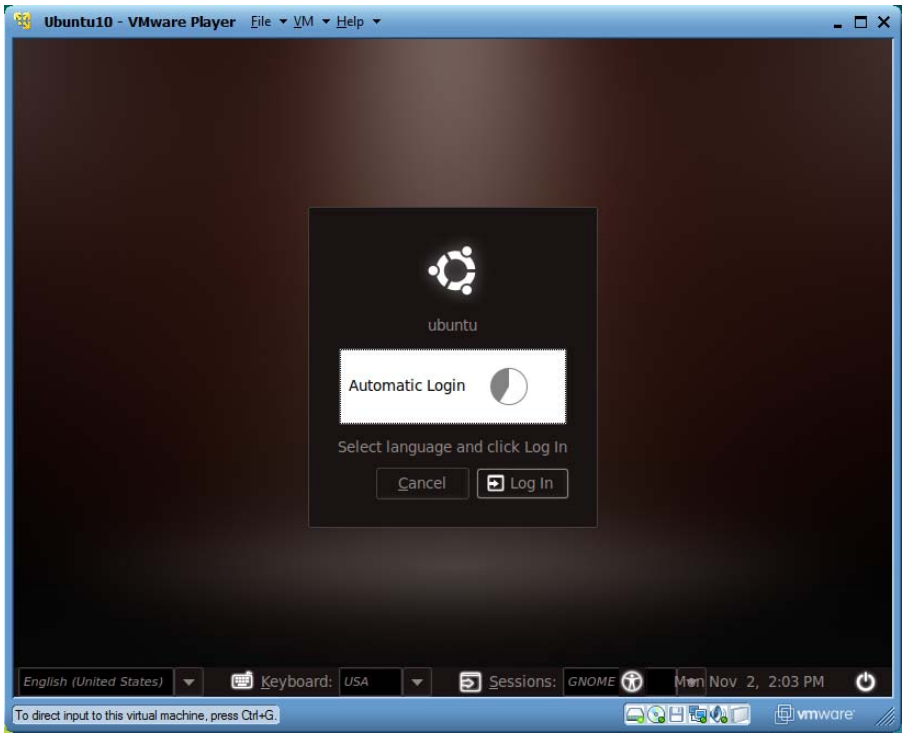
©Copyright 2009 Prof. Lixin Tao
and you will see the following “Login Screen Settings” window:

Click on the *Unlock* button and you will see the following “Authenticate” window:

Enter 12345678 as your password and click on the *Authenticate* button, the “Authenticate” window should disappear (if you are running *VMware Player* earlier version, you may need to close this “Authenticate” window manually). You may need to click on the *Unlock* button again to be able to change the settings as below:




Here I have chosen to give the user 10 seconds to type in a different user name before the VM automatically log in as “user”. Click on the *Close* button and you are done with this task. When you restart your VM, you will see a login screen as below:




You will be logged in as “user” automatically if you don’t type another user name within 10 seconds.

## 2.4 Changing Mouse Focus and Exchanging Files between VM and PC

At any time, if you need to use the mouse cursor in the VM, just click anywhere in the VM window; and if you need to use the mouse outside of the VM window, type key combination “Ctrl+Alt”.

There are four ways for exchanging data between your VM and your hosting PC.

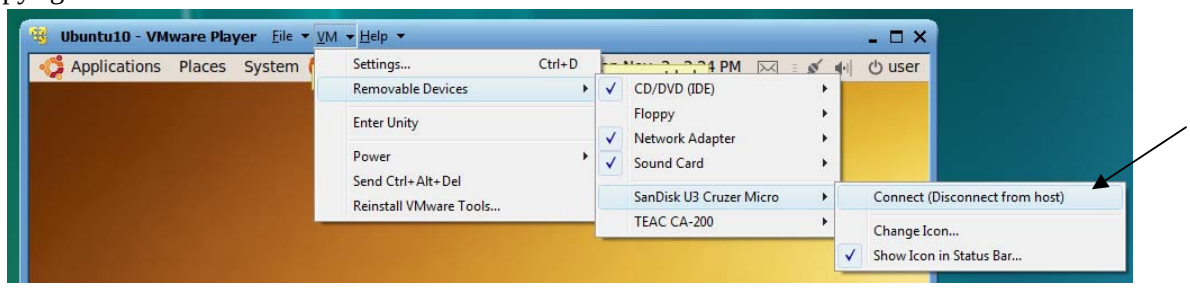
### *2.4.1 Dragging-and-Dropping*

At any time you can use the left mouse button to drag-and-drop files and folders between your physical PC and your VM. This is the simplest way to share files between your PC and VM.

### *2.4.2 Using a USB Portable Disk.*

If you are working in the VM window and you plug in a USB device, the USB device will be recognized by your VM. On the other hand if you are working outside of the VM, then your PC will take over the USB device. You can also use *VMware Player*’s menu “VM|Removable Devices” to disconnect a USB disk drive, DVD drive or floppy drive from the VM so it could be recognized and used by the PC, or disconnect a drive from your PC so it could be recognized and grabbed by your VM.

©Copyright 2009 Prof. Lixin Tao

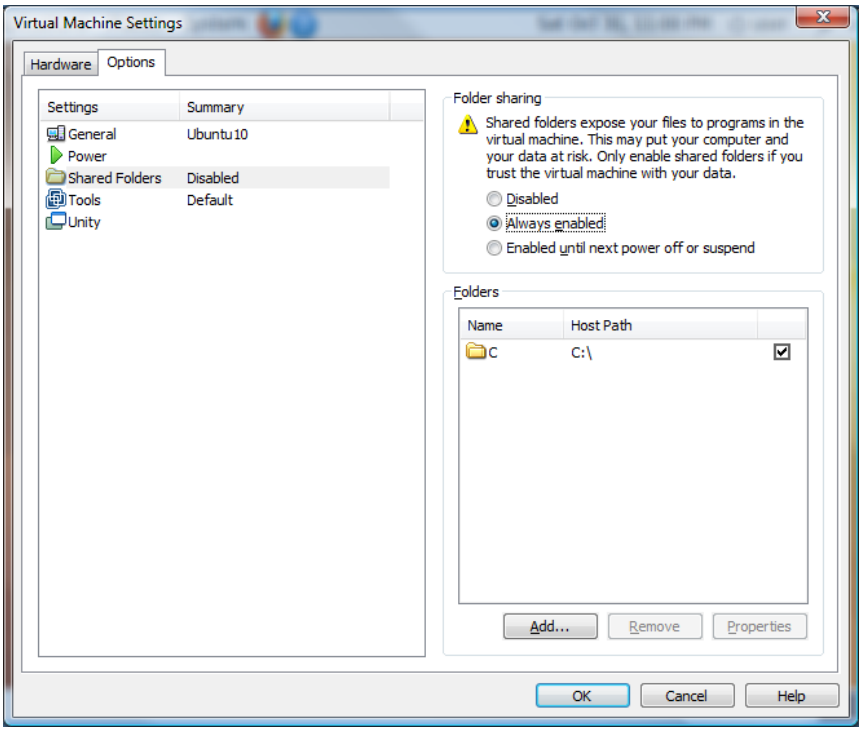



Therefore we could use a USB disk as a bridge for exchanging files between the PC and the VM.

By the way in the last screen capture, you can also see a *VMware Player* menu item "Send Ctrl+Alt+Del", which is how we generate the Ctrl+Alt+Del key combination, an important key combination for launching the task manager of *Windows* VMs.

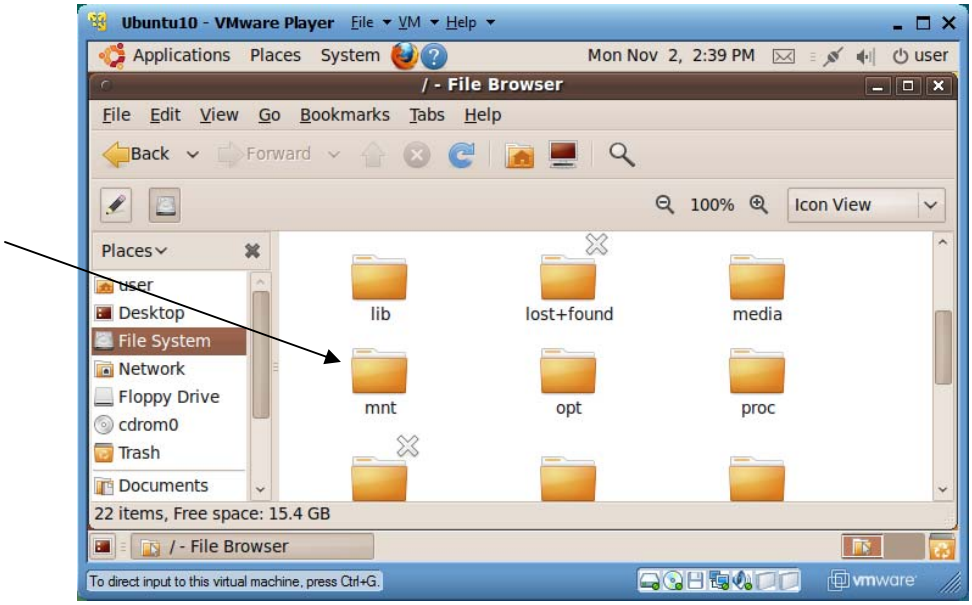
### *2.4.3 Shared Folders*

I have set up a shared folder "/mnt/hgfs/C" on this VM and you can use it to access your PC's entire "C:\" drive. When you launch the VM for the first time, you may need to enable it by clicking on *VMware Player* menu item VM|Settings…", choosing the "Options" tab, selecting "Shared Folders" in the left pane, and making sure that the checkbox for "Always enabled" is checked and the folder "C" for "C:\" is checked, as shown below:



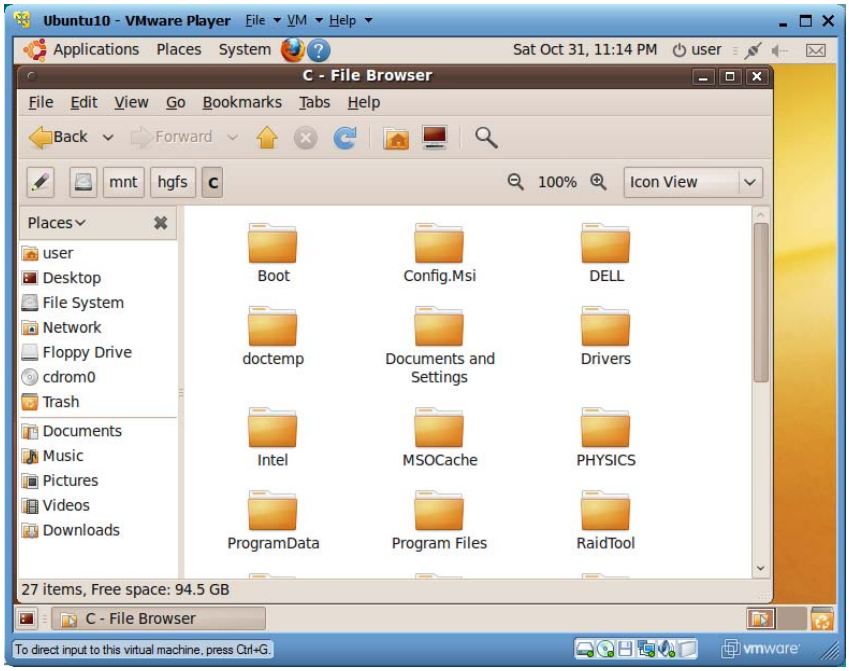

If you click on Ubuntu menu item "Places|Computer",

you will see the following file browser window:




If you double-click on “mnt” (mount), then the embedded “hgfs”, you will see a folder “C”. Double-click on it and you will have access to all of your PC’s drive “C:\”, as shown below. If you cannot see folder “C” under “hgfs”, you need to enable this shared folder as described earlier in this subsection.

©Copyright 2009 Prof. Lixin Tao




### *2.4.4 Coping-and Pasting Text between VM and PC*

You can use the familiar copy-and-past to copy text between your PC and VM. Remember the following exceptions:

- In a *Ubuntu* terminal window, you can select text by just highlighting the text.
- In a *Ubuntu* terminal window, you use key combination Ctrl+Shift+v to paste text into it, instead of familiar key combination Ctrl+v.
- For other *Ubuntu* tools you can try Ctrl+v and Ctrl+Shift+v to see which combination work for pasting text.

## 2.5 Using Basic *Ubuntu* Tools

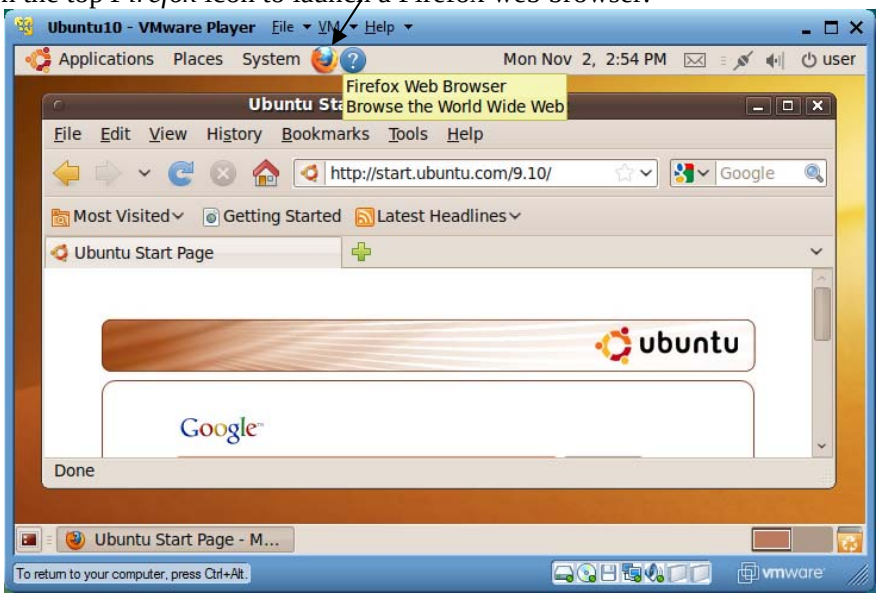
### *2.5.1 Launching a Terminal Window*

To open a terminal window to run some Linux commands, you can double-click on menu item “Applications|Accessories|Terminal”:

You can open multiple terminal windows.

### *2.5.2 Launching a Firefox Web Browser*

You can click on the top *Firefox* icon to launch a Firefox web browser.

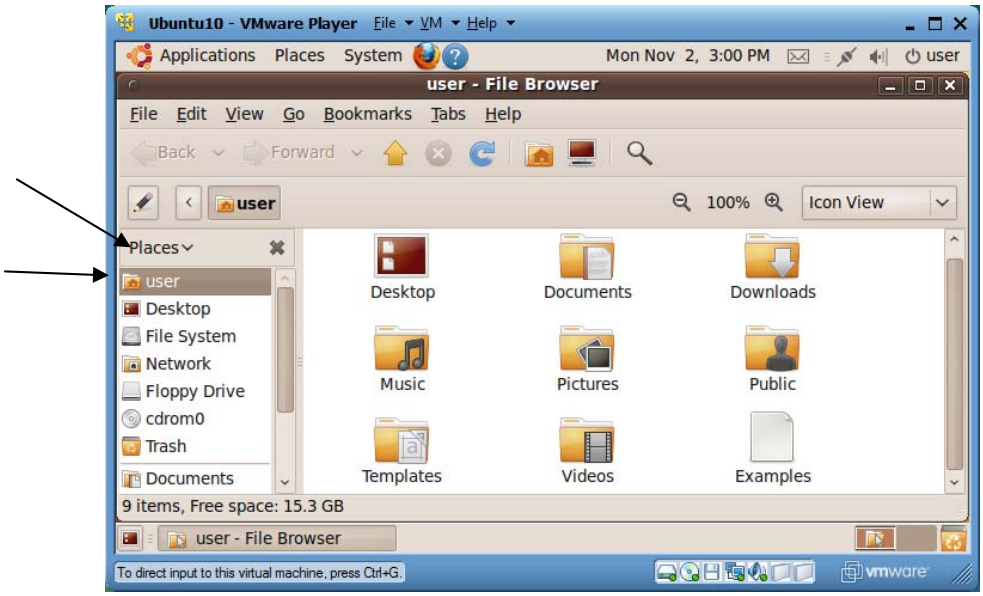



### *2.5.3 Launching a Nautilus File Browser*

*Ubuntu* menu “Places” lets you browse files in various places of the file system with a *Nautilus* file browser.

©Copyright 2009 Prof. Lixin Tao

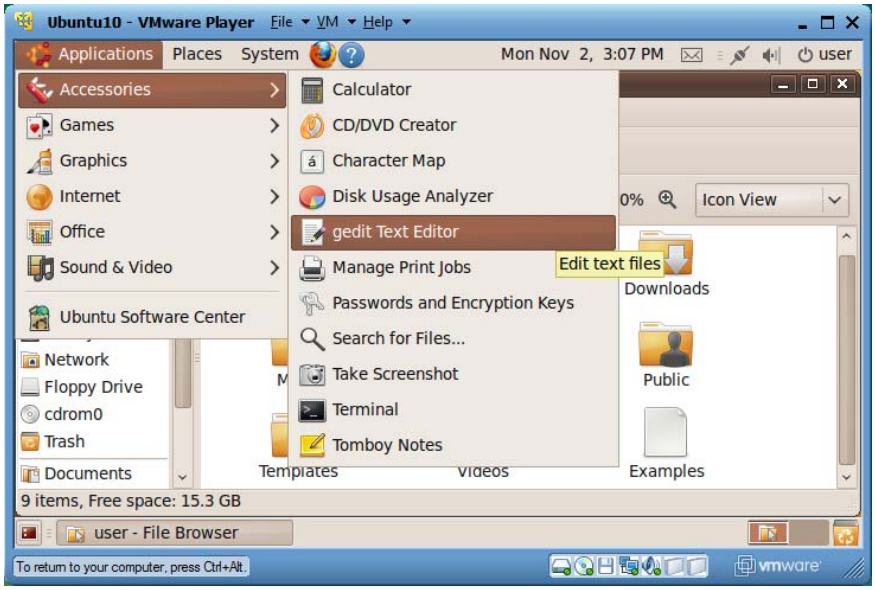
For example, if you double-click on the "Places|Home Folder" menu item, you will see the following file browser:




In the left "Places" pane, "user" is the current user *user*'s home folder "/home/user", which is also represented by symbol ~ in file/folder paths; and "File System" is the entire Linux file system "/". Folder "/home/user/Desktop" holds all items on your Ubuntu desktop. When you use Firefox web browser to download files, the downloaded files will be saved in folder "/home/user/Downloads".

### *2.5.4 Launching a Text Editor*

You can use *Ubuntu* menu item “Applications|gedit text Editor” to launch a *gedit* text editor.




### *2.5.5 Using Synaptic Package Manager*

*Ubuntu* menu “System” contains important administrator tools for managing the system. Menu item “System|Administration|Synaptic Package Manager” allows you to use a GUI to install, update or remove software packages (in this document we will use a lower-level command tool “aptitude” for performing the same functions).

©Copyright 2009 Prof. Lixin Tao

### *2.5.6 Customizing Ubuntu Application Menu*

Menu item “Preferences|Main Menu” lets you customize the top-left *Applications* menu.

## 2.6 Finding Out Your Gateway/Router, PC and VM's IP Addresses

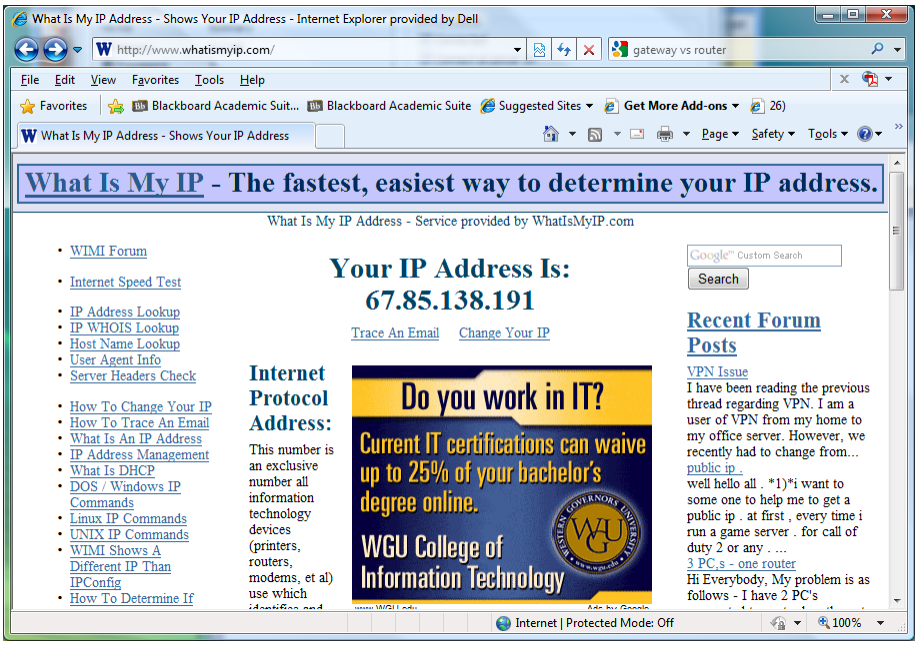
Your home gets *Internet* access through a gateway connected to your phone or cable service line. Your gateway has a unique IP address visible on the Internet, and this address by default is dynamically assigned when you power up the router. If you run a home business on the Internet, you may want to upgrade to a static IP address service so your clients can easily access your online services. Your gateway also has a unique local area network (LAN) IP address (not visible outside of your home), which is normally called the *gateway address*. Your gateway normally also functions as a wired/wireless router to support communications among the PCs in your LAN as well as accessing the Internet by your LAN PCs. Each PC in your LAN has a unique local IP address of form 192.168.x.y. Each message (IP packets) on the network has its destination IP address. An Internet message for any of your local PCs will use your gateway's external IP address as its destination, and the router will use a routing table to forward the message to your particular PC which may have requested for that message. When one of your PCs sends a message to another PC in your LAN, the router will also use its routing table to forward the message to the right PC. All PCs in your LAN as well as the router must use the same subnet mask, normally in form of "255.255.255.0", and their local IP addresses must have the same value for the first three integers, like 192.168.0.x, where x means any integer. To put it simple, a subnet mask defines an IP address pattern for all PCs in the same LAN, and only PCs whose IP address follows this pattern is considered part of the same LAN. You could have multiple LANs in your home if your PCs use multiple subnet mask values.

Your router normally also runs a DHCP (Dynamic Host Configuration Protocol) server. When a PC is connected to your router, the PC will send over its network card's MAC (Media Access Control) address, which is supposed to be unique world-wide with high probability. The router will assign a unique local IP address for this PC and send this address back to the PC. A local IP address is of form "192.168.x.y" and it can only be used in your LAN for PCs to access each other.

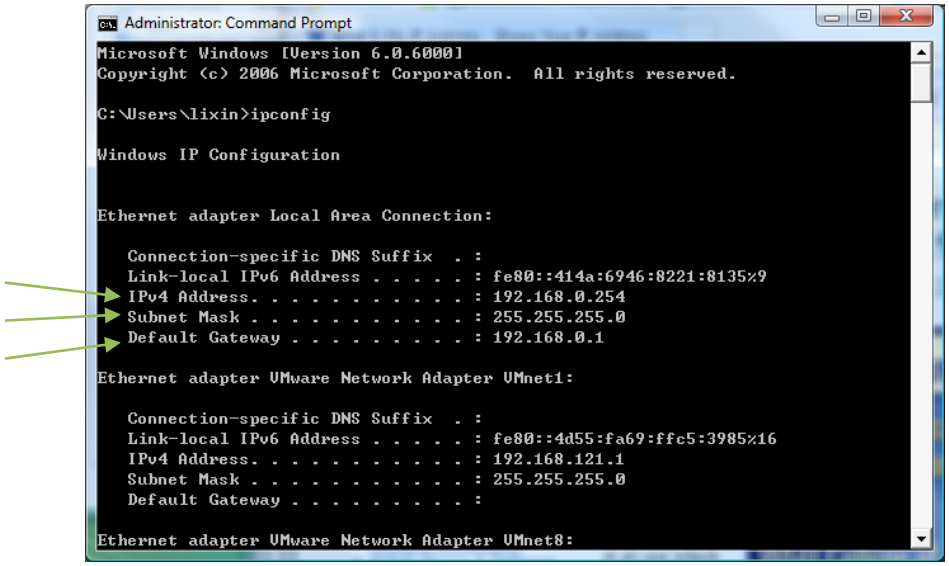
The above is a concise review, not accurate introduction, of the main networking concepts relevant to our discussion. You can search textbooks or the web for more detailed explanations.

To find out what is your Internet gateway/router's IP address, start a web browser in your PC, and visit address http://www.whatismyip.com. The following is a sample screen showing that my router's current external IP address is 67.85.138.191.

©Copyright 2009 Prof. Lixin Tao

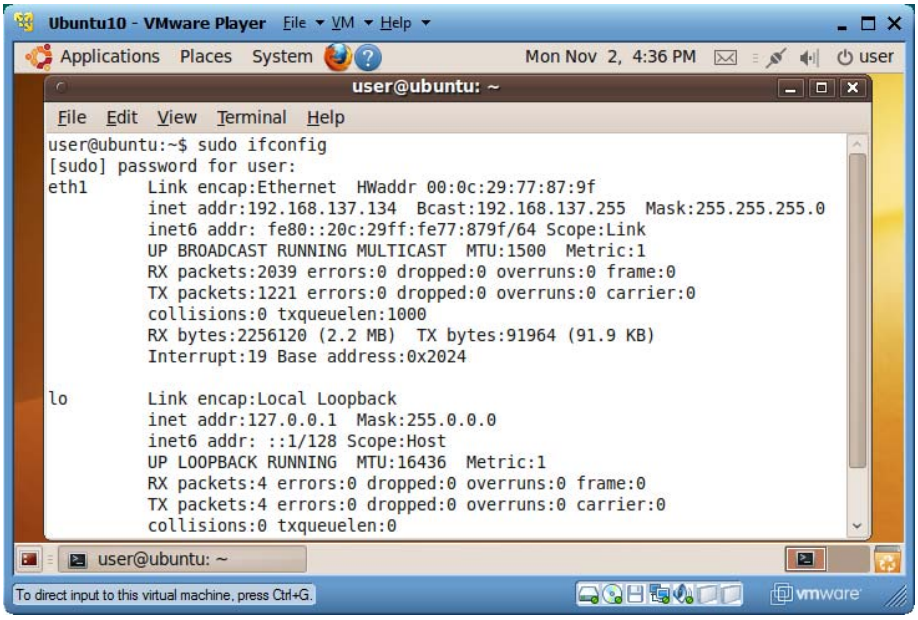



To find out the local IP address of your *Windows* PC or VM, run "Start|All Programs|Accessories|Command Prompt" to launch a DOS window, and run command "ipconfig" in the DOS window. For example, the following is part of my *ipconfig* session, and it tells me that my local IP address is 192.168.0.254 (IPv4 Address), my subnet mask is 255.255.255.0, and my default gateway has IP address 192.168.0.1. Ignore the information for VMnets because they are for the virtual network adapters.




To find out the local IP address of your *Linux* PC or VM, start a terminal window and run command "sudo ifconfig". You will be asked to enter your Linux password. After you enter your password, you will see a lot of information about your LAN, including your local IP address (inet addr: 192.168.137.134) and subnet mask (Mask: 255.255.255.0). You only need to read the "eth1" section. The "lo" section is information for networking within this PC or VM. All *Windows* and *Linux* computers have a special local loopback IP address 127.0.0.1. If a message is sent to this address, it will never leave the computer that

generates this message, and the message will be sent back to the same computer. This address is useful to test servers on the same computer. For example, if we run a web server on a computer, we may want to test it by launching a web browser on the same computer. You can now enter 127.0.0.1 as the URL of the server computer. For your convenience most computers have defined a domain name *localhost* as nickname for 127.0.0.1.




## 2.7 Changing Network Connection Method

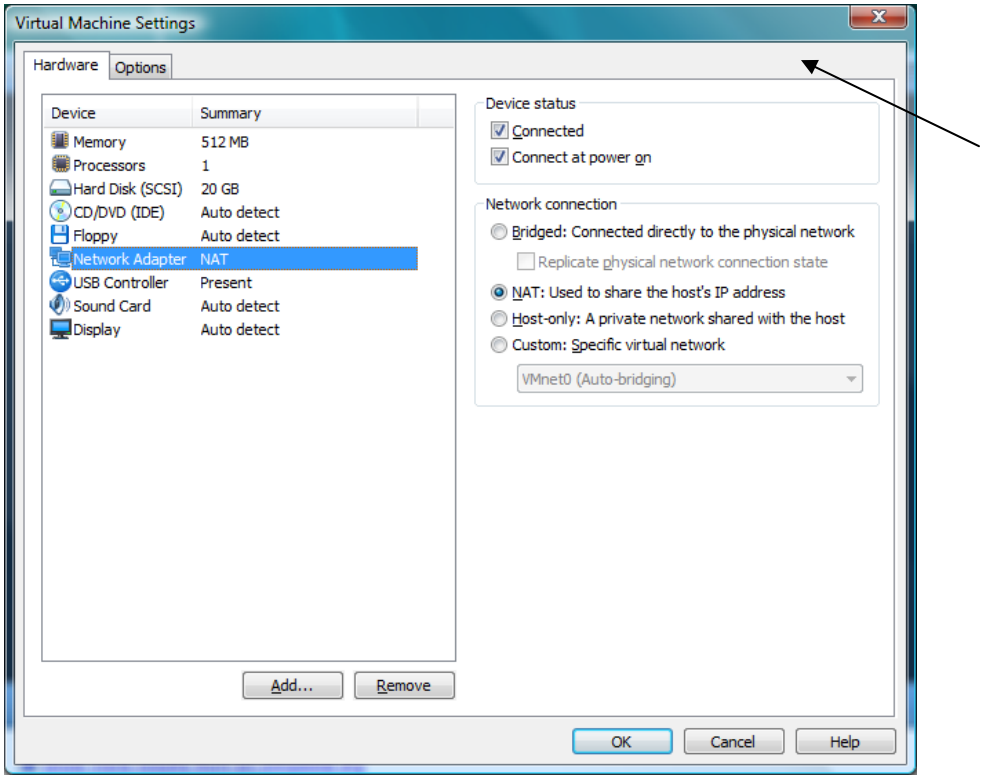
Before reading this section, you should first read subsection “Finding Out Your Gateway/Router, PC and VM’s IP Addresses”.

There are three popular methods to connect your VM to your host PC and the Internet:

1. Bridged. With this method, your VM will contact directly your local DHCP (Dynamic Host Configuration Protocol) server and apply for a unique local IP address in your LAN (local area network). Your VM can then access the *Internet* and all resources available in your LAN, and other PCs in your LAN can access resources on your VM. For example, if you run a web server on this VM, any other PCs or VMs in your LAN can access it with this VM’s local IP address.
2. NAT (network address translation). With this method, your VM can access the Internet as if it is its host PC (the VM share the host PC’s local IP address), access other VMs running on the same host PC with NAT method, and its host PC can access services or data on this VM. But the other PCs in your LAN will not be able to access this VM.
3. Host-only. This method is the same as NAT except the VMs cannot access the Internet. The VMs running on the same host PC and using the same host-only method can access each other’s services.

You can change your VM’s networking connection method with *Ubuntu* menu item “VM|Settings…”.
The following “Virtual machine Settings” window will pop up. Select its *Hardware* tab, then select “Network Adapter” in the left pane. No wyou can choose which network connection method to use in the right pane. This example uses the NAT method. After you change the method and click on the *OK* button,

©Copyright 2009 Prof. Lixin Tao
you will lose network access for a few seconds before it comes back. Sometimes you need to restart the VM to get network access back. The VM will remember its network connection method.




## 2.8 Running a Cluster of VMs on a Single Computer

Before reading this section, you should first read subsection “Changing Network Connection Method”.

To run multiple VMs on a single PC concurrently, following the following steps:

1. Make sure that each VM has its unique virtual network card MAC (Media Access Control) address. You can do so by moving the VM to a new file system location or changing the VM folder name, and answer “I copied it” when you launch the new copy of the VM.
2. Use a text editor to modify the value of “memsize” (memory size for the VM) so the total memory size used by your all of your VMs to run together is at least 512 MB less than your total available physical memory (512 MB for your PC to use). Each VM should have 256 MB or more. The exact minimal memory size for each VM depends on type of OS for the VM and what applications have been installed on it, so you need to test the best memory sizes for the VMs. You should not try to run multiple VMs concurrently unless you have 2 GB or more physical memory.
3. Launch each of the VMs, and make sure that all the VMs to run concurrently on the same PC use the same network connection method: bridged, NAT or host-only (the host-only method will not provide Internet access to the VMs). Read subsection “Changing Network Connection Method” for their differences.

4. When each VM runs, use "sudo ifconfig" on Linux VM or "ipconfig" on Windows to find out the local IP address for each of the VMs in the cluster. Refer to subsection "Finding Out Your Gateway/Router, PC and VM's IP Addresses" for details.
5. Now you are ready to treat the cluster as a local area network and emulate a complete enterprise IT infrastructure.

By now you have learned the essence of what is very hot in the industry now: *IT system virtualization technologies*.

## 2.9 Learning Basic Linux Commands

Now you can launch terminal windows, you can use the rich online tutorials to learn and practice *Linux* commands. The following are just a few starting points for you to gain Linux command line proficiency:

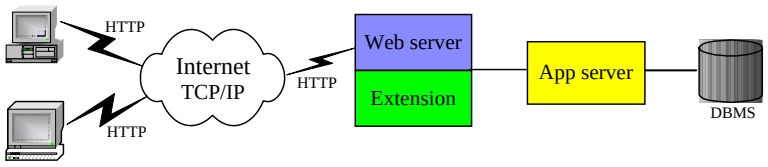
- http://www.linfo.org/command_line_lesson_1.html
- http://www.tuxfiles.org/linuxhelp/
- http://linuxcommand.org/learning_the_shell.php
- http://linuxcommand.org/
- http://www.ee.surrey.ac.uk/Teaching/Unix/
- http://www.stat.tamu.edu/~dahl/teaching/605/1.21/shell-tutorial/html/index.html

In the next edition of this tutorial I will add sections on basic concepts in Linux and the introduction of each command used in this tutorial.

## 2.10 Technical Summary of *Ubuntu* V9.10 and Installed Applications

### *2.10.1 Major Installed Web Architecture Components*

A typical web application involves four tiers as depicted below: web browsers on the client side for rendering data presentation coded in HTML, a web server program that generates data presentation in HTML, an application server program that computes business logic and generates data, and a database server program that provides data persistency. The three types of server programs may run on the same or different server machines.




| Tier 1 | Tier 2 | Tier 3 | Tier 4 |
|---|---|---|---|
| Web browser | Web server | App Server | Database |
| Firefox | Apache + Perl +PHP<br>Tomcat<br>GlassFish | GlassFish | MySQL<br>Derby |

©Copyright 2009 Prof. Lixin Tao
Our *ubuntu10* VM has installed one or more components in each of the four web tiers. *Firefox* works as a web browser in tier 1 on the client machine. *Apache* with *Perl* and *PHP* CGI extensions, *Tomcat*, and *GlassFish* support tier 2 as web servers. *GlassFish* is also an enterprise-quality application server featuring EJB technologies. The VM has two database server systems installed: *MySQL* and *Derby*. For your reference, IBM's *WebSphere* and BEA's *WebLogic* (now owned by Oracle) are both organized like *GlassFish* supporting both web server and application server functions.

Our *ubuntu10* VM supports both *NetBeans* and *Eclipse*, the most popular IDEs, for developing applications on each of the above four web tiers.

### *2.10.2 Installation Credentials and Technical Summary*

**Ubuntu V9.10:**

Super User: root Password: 12345678
Normal User: user Password: 12345678

**Java JDK V6.6.0_16**

Installation folder: /home/user/tools/jdk1.6.0_16
Shared by all users through "/etc/profile"

**Tomcat Web Server V6.0.20:**

Installation folder: ~/tomcat
Web doc root: ~/tomcat/webapps
Home page: http://localhost:8080
Tomcat starts automatically at system boot-up time.
To manually start: run "tomcat-start"
To manually stop: run "tomcat-stop"

**Apache Web Server V2.2:**

Installation folder: /etc/apache2
Web doc root: ~/www (/var/www)
Home page: http://localhost
Apache starts automatically at system boot-up time.
To manually restart Apache, run "sudo apache2ctl restart"
Apache supports Perl, PHP and MySQL.
Perl script folder: ~/www/cgi-bin

**MySQL Database Server V5.1:**

Username: root
Password: 123456
Launch MysQL admin console: run "mysql -u root -p123456"

**Eclipse IDE V1.2.1.20090918-0703 (Galileo SR1):**

Installation folder: /home/user/tools/eclipse
To start Eclipse, run "eclipse", or use menu item "Applications|Programming|Eclipse"
Default project home: /home/user/EclipseWorkspace
Servers supported by Eclipse: Tomcat and GlassFish v2.1

**NetBeans IDE V6.7.1:**

Installation folder: /home/user/tools/netbeans-6.7.1
To start NetBeans, run “netbeans”, or use menu item “Applications|Programming|NetBeans IDE”
Default project home: /home/user/NetBeansProjects
Servers supported by NetBeans: Tomcat and GlassFish v2.1

**GlassFish Application Server V2.1 (including web server):**

Installation folder: /home/user/tools/SUNWappserver
URL: http://localhost:8081 https://localhost:8181
Admin console: http://localhost:4848
Username: admin
Password: adminadmin
Start GlassFish server: run “glassfish-start”
Stop GlassFish server: run “glassfish-stop”

**Derby Database Server V10.4.2.1**

Installation folder: /home/user/tools/SUNWappserver/javadb
Start Derby server: run “derby-start”
Stop Derby server: run “derby-stop”
Launch Derby admin console: run “ij”

**Drupal Contents Management System V6.14:**

URL: http://localhost/drupal
Doc root folder: ~/www/drupal
Username: root
Password: 123456

**Sample Tomcat web applications:**

- http://localhost/examples/servlets
- http://localhost/examples/jsp
- http://localhost/survey
- http://localhost/surveyJsp
- http://localhost/surveyCss
- http://localhost/surveyJsf
- http://localhost/demo
- http://localhost/html
- http://localhost/tripler
- http://localhost/testCookie
- http://localhost/bookstoreServlet
- http://localhost/bookstoreJsp
- http://localhost/axis
- http://localhost/bareJsp
- http://localhost/guessNumber

**Sun Java Tutorials:**

**Java Tutorial 2009-09-23**

Base folder: /home/user/JavaTutorials/JavaTutorialls

©Copyright 2009 Prof. Lixin Tao

Tutorial: use web browser to visit "/home/user/JavaTutorials/JavaTutorial/index.html" to read the Tutorial in web browser.

**Java EE 5 Tutorial V1.0_05**

Base folder: /home/user/JavaTutorials/javaeetutorial5

Tutorial: use web browser to visit "/home/user/JavaTutorials/javaeetutorial5/doc/index.html" to read the tutorial in web browser.

Tutorial PDF version is available at *http://java.sun.com/javaee/5/docs/tutorial/doc/JavaEETutorial.pdf.*

**First Cup Tutorial for Java EE V2.1**

Base folder: /home/user/JavaTutorials/firstcup

Tutorial: use web browser to visit "/home/user/JavaTutorials/firstcup/doc/index.html" to read the tutorial in web browser.

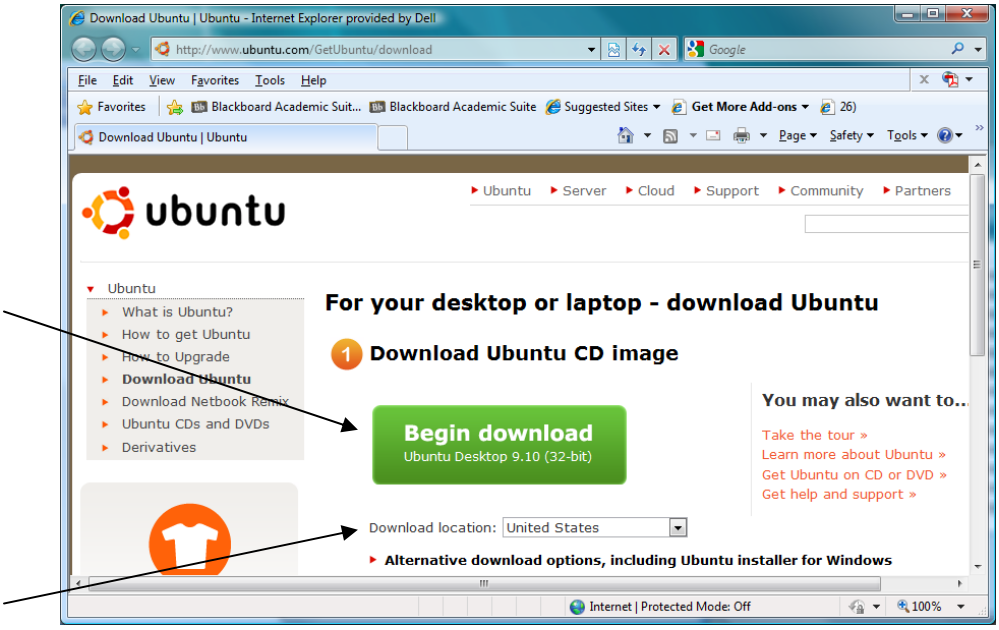
PDF version of the tutorial is "/home/user/JavaTutorials/firstcup/doc/firstcup.pdf".

# 3
# Setting Up a Basic *Ubuntu* Virtual Machine

In this section we create a basic *Ubuntu* V9.10 virtual machine using *VMware Player* v3.0. For avoiding confusion, rename the VM *ubuntu10* downloaded from my web site in Section 2 as "C:\VM\ubuntu10completed".

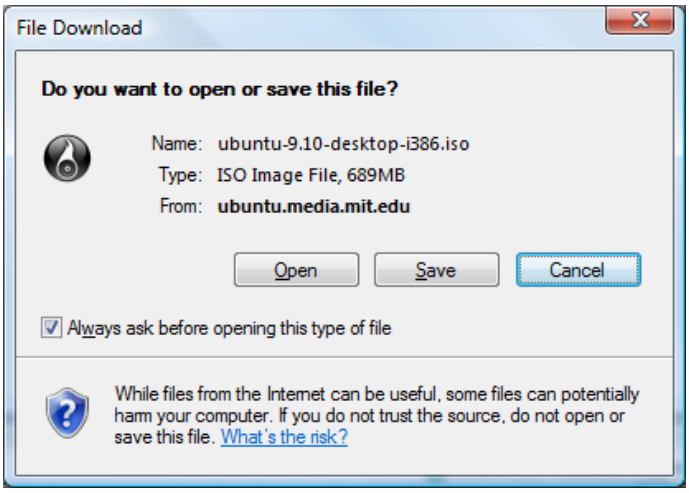
## 3.1 Downloading *Ubuntu*

- Visit http://www.ubuntu.com/GetUbuntu/download. Select United States as the download location, and then click on the "Begin download Ubuntu Desktop 9.10 (32-bit)" button.




- If you get browser security warning, right-click on the security warning immediately below the tools menu, and choose "Download File…". In either case you will see the following "File Download" window.

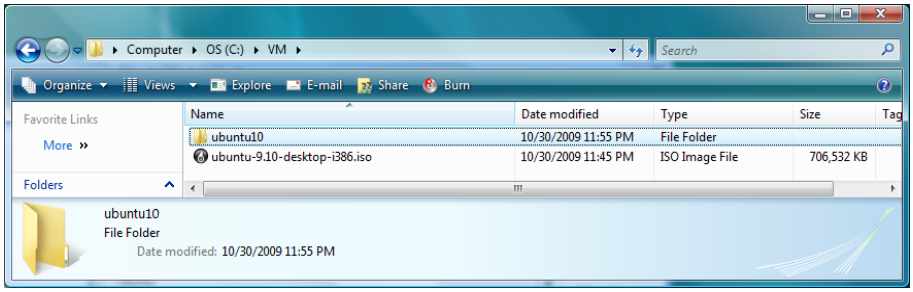
©Copyright 2009 Prof. Lixin Tao




- Save file "ubuntu-9.10-desktop-i386.iso" in "C:\VM". The file has 689 MB and may take 20 minutes to download.

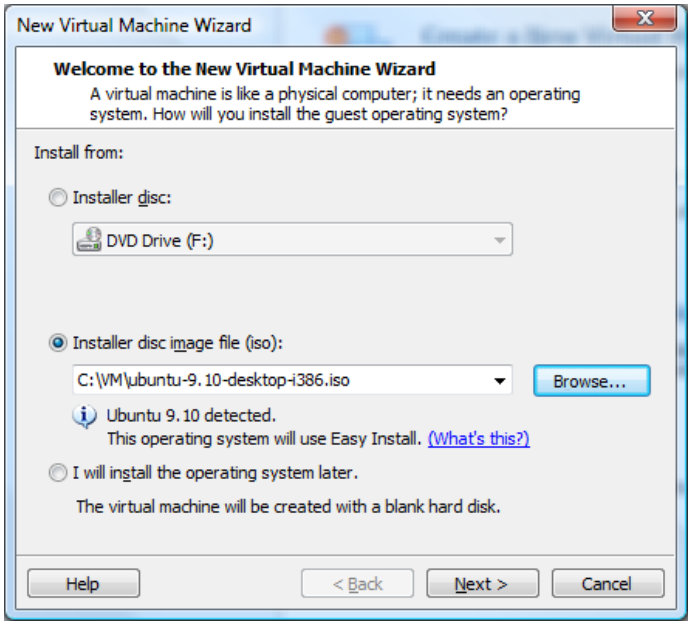
## 3.2 Creating a *Ubuntu* Virtual Machine

- In "C:\VM", create a new folder "ubuntu10".



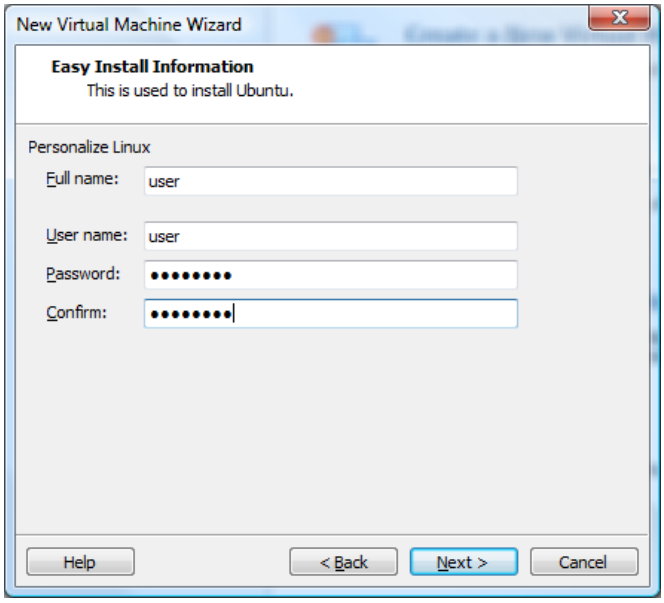

- Launch *VMware Player*.

- Click on the link for “Create a New Virtual Machine”.

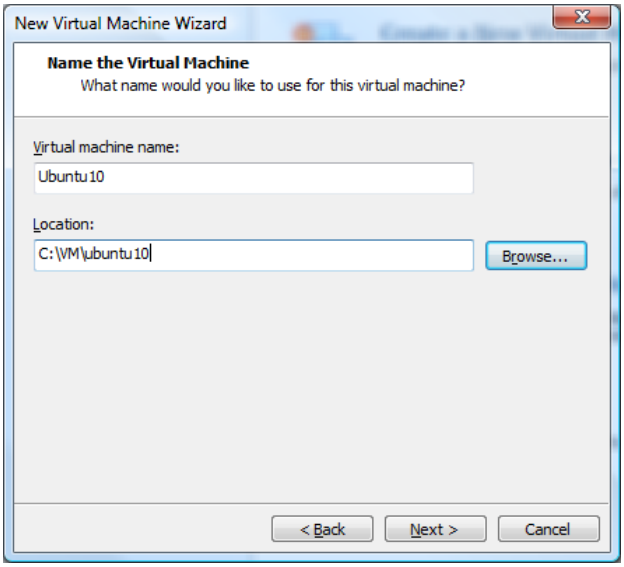



- Check for “Installer disc image file (iso), and browse for your downloaded *Ubuntu* v9.10 iso file (C:\VM\ubuntu-9.10-desktop-i386.iso). Click on the *Next* button.

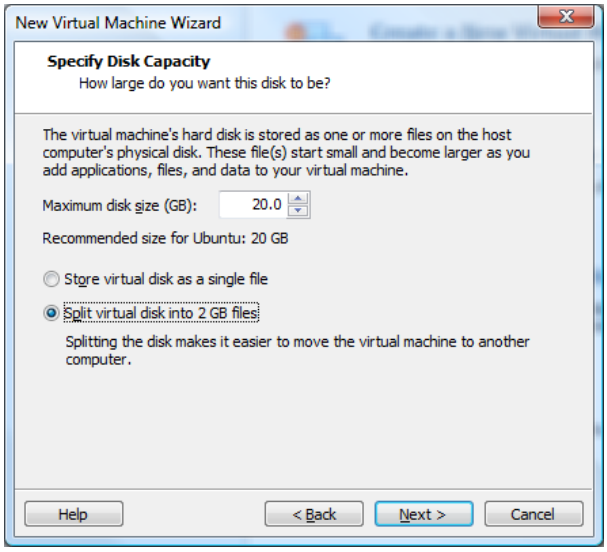
©Copyright 2009 Prof. Lixin Tao



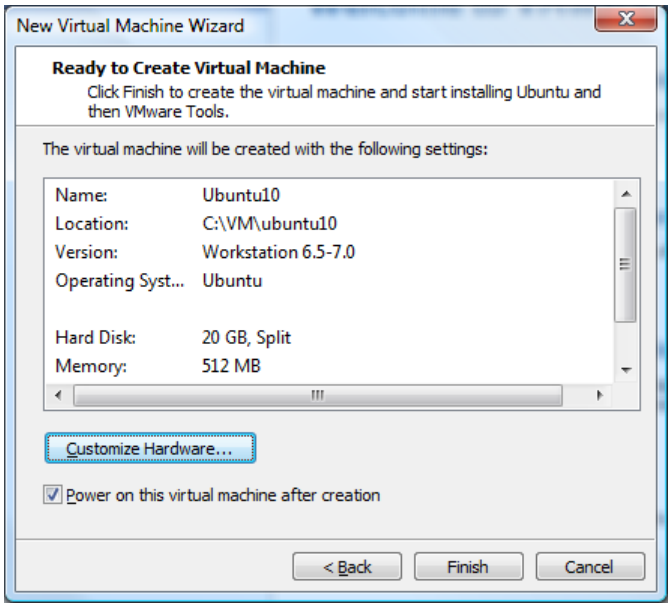

- In the popup “New Virtual Machine Wizard” window, enter “user” for full name and user name, and 12345678 for password and confirm. Then click on the *Next* button.




- Use “Ubuntu10” as the virtual machine name, and “C:\VM\ubuntu10” as the location value. Then click on the *Next* button.

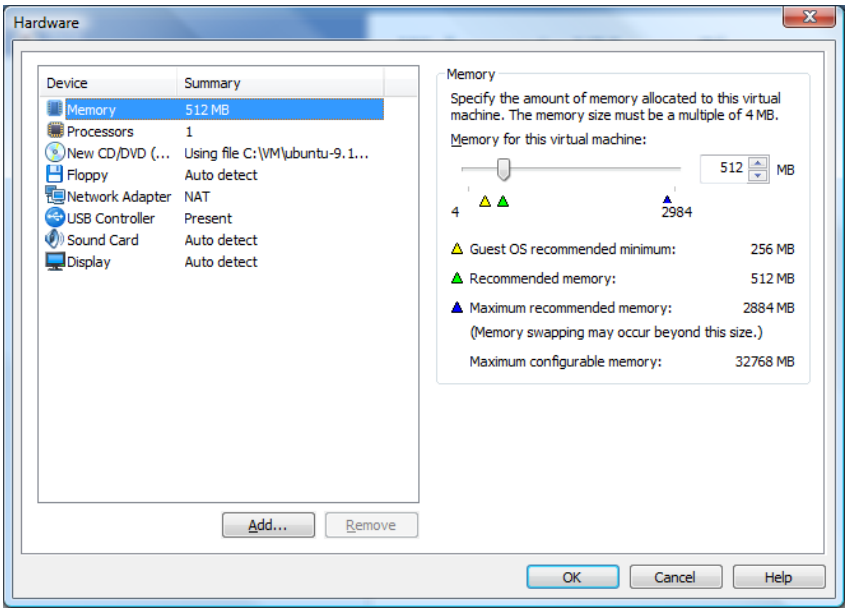



- Enter 20 for maximum disk size, and check for “Split virtual disk into 2 GB files” so we could later easily save the VM on DVD disks. Then click on the *Next* button.

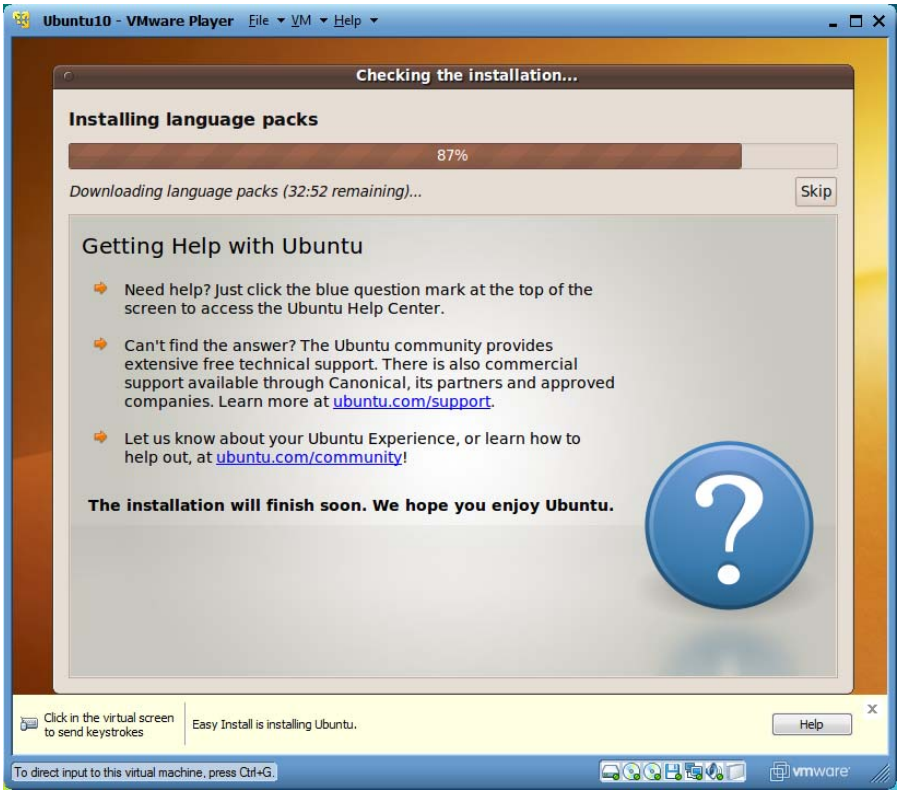



- Review the settings for the new VM. Click on the “Customize hardware…” button to review what you can further customize. Notice that by default the new VM uses 512 MB physical memory. Since your PC’s OS also needs physical memory, your PC needs to have one GB memory to work well. If you have more than one GB memory, you could increase the VM’s memory size so it could work smoother. If you plan to run multiple VMs at the same time, make sure the total memory used by these VMs and your PC would not exceed your available physical memory size. You may let a VM work with only 256 MB memory.

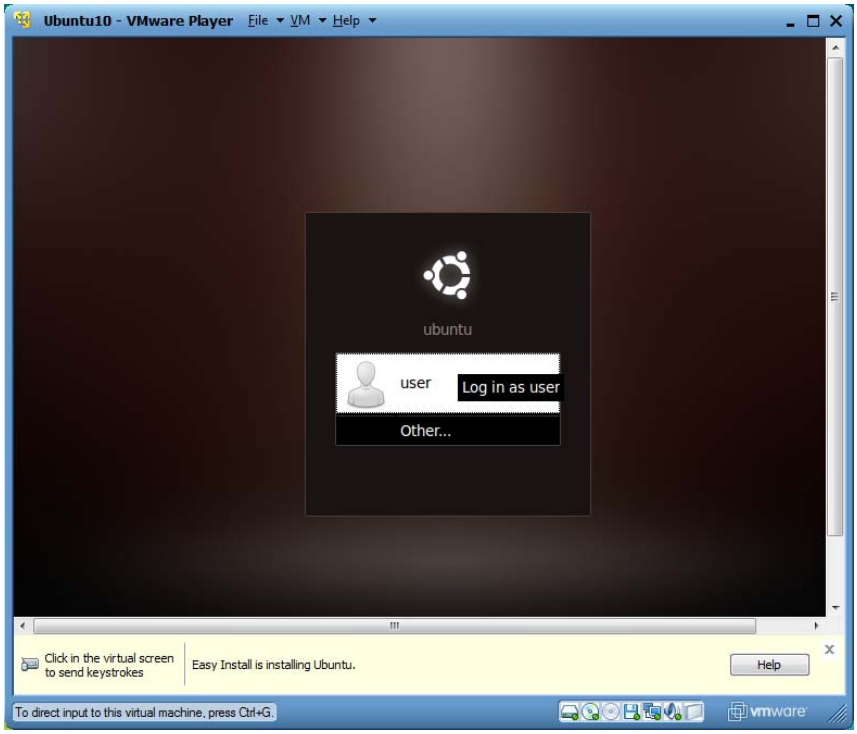
©Copyright 2009 Prof. Lixin Tao



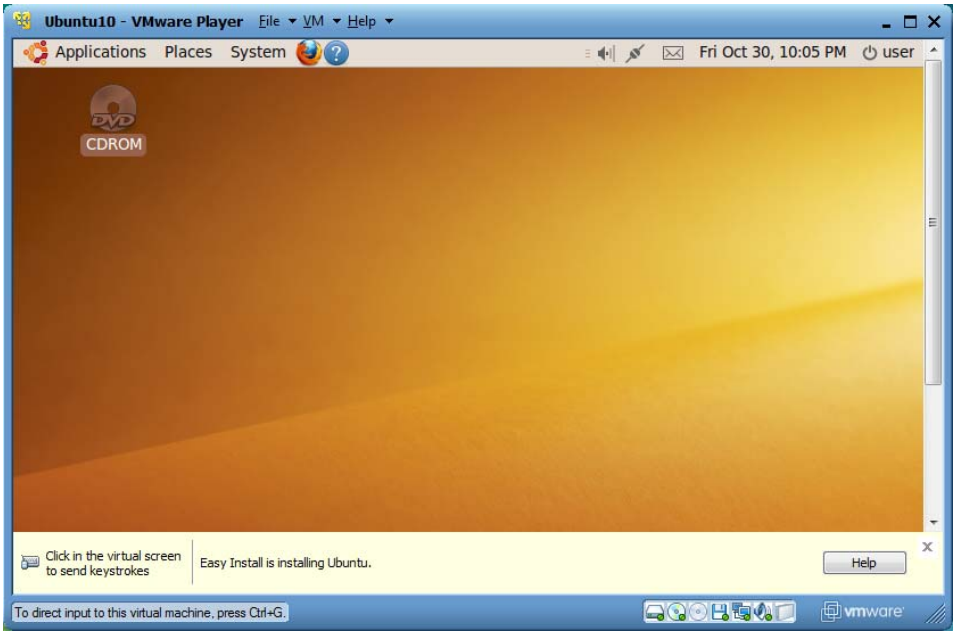

- Click on the OK button to exit the "Hardware" window. Click on the "Finish" button to start the Ubuntu installation process.
- If you are asked for permission to download *VMware Tools*, give the permission.




- The installation process may take one hour. It doesn't need your intervention. You can just take the time to do something else. At the end, Ubuntu will reboot automatically.

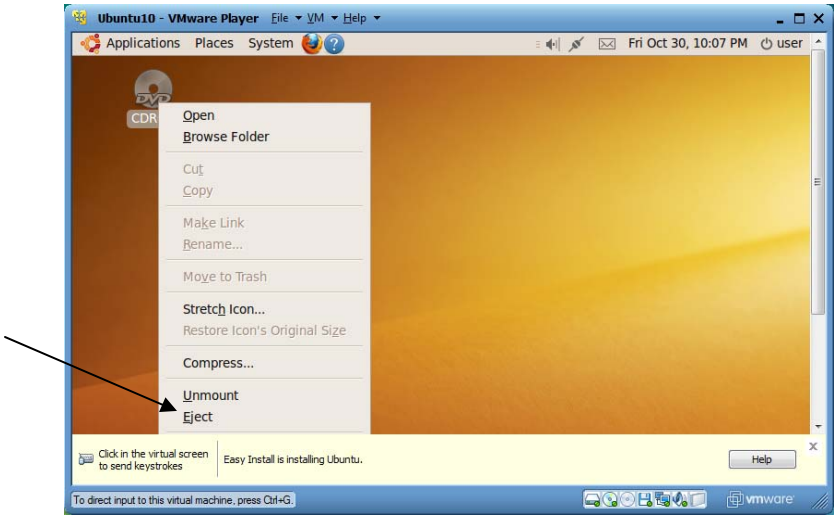



- Click on “user”, and enter 12345678 as password.

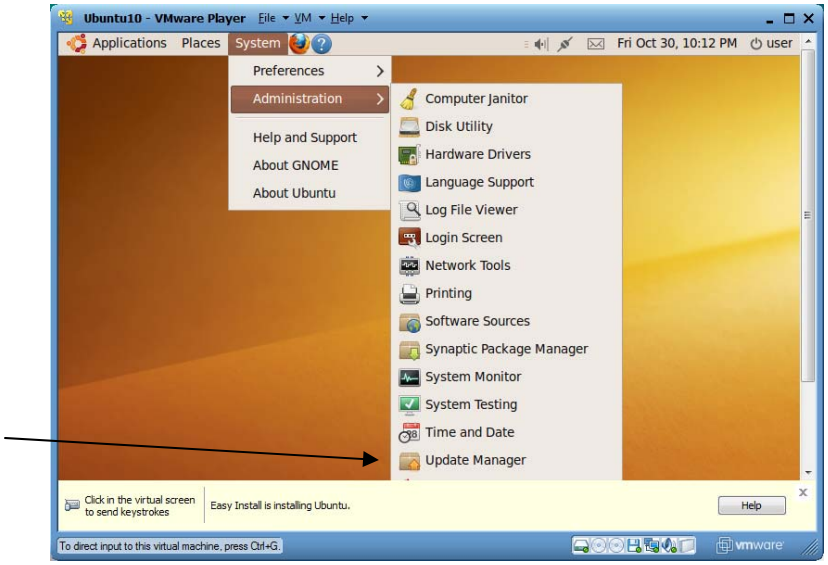



- Right-click on the CDROM icon on the Desktop, and choose “Eject on the popup menu to eject the *Ubuntu* installer CD.

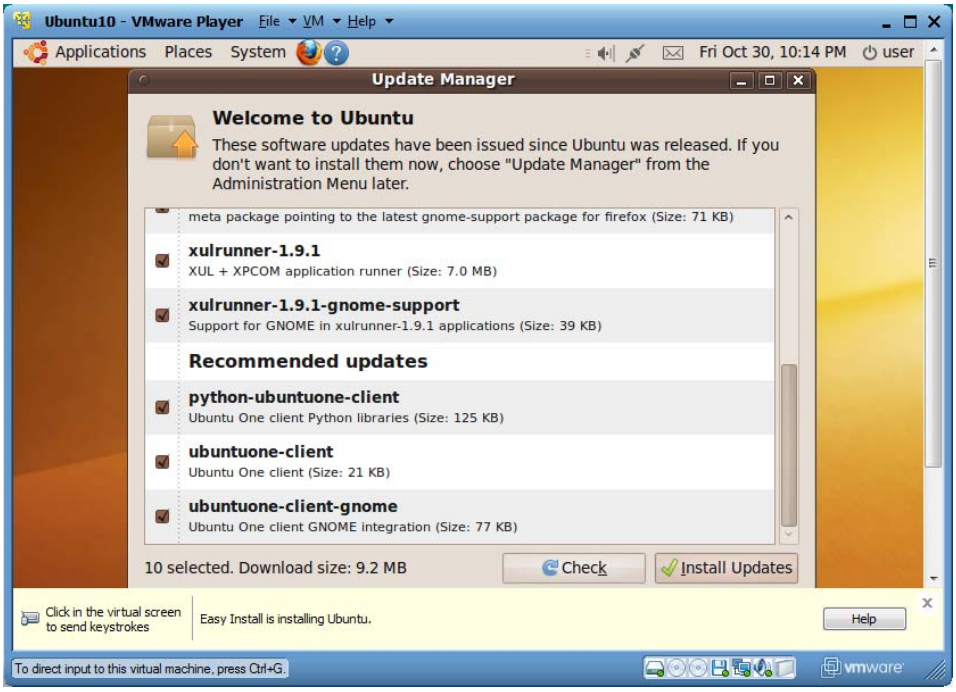
©Copyright 2009 Prof. Lixin Tao



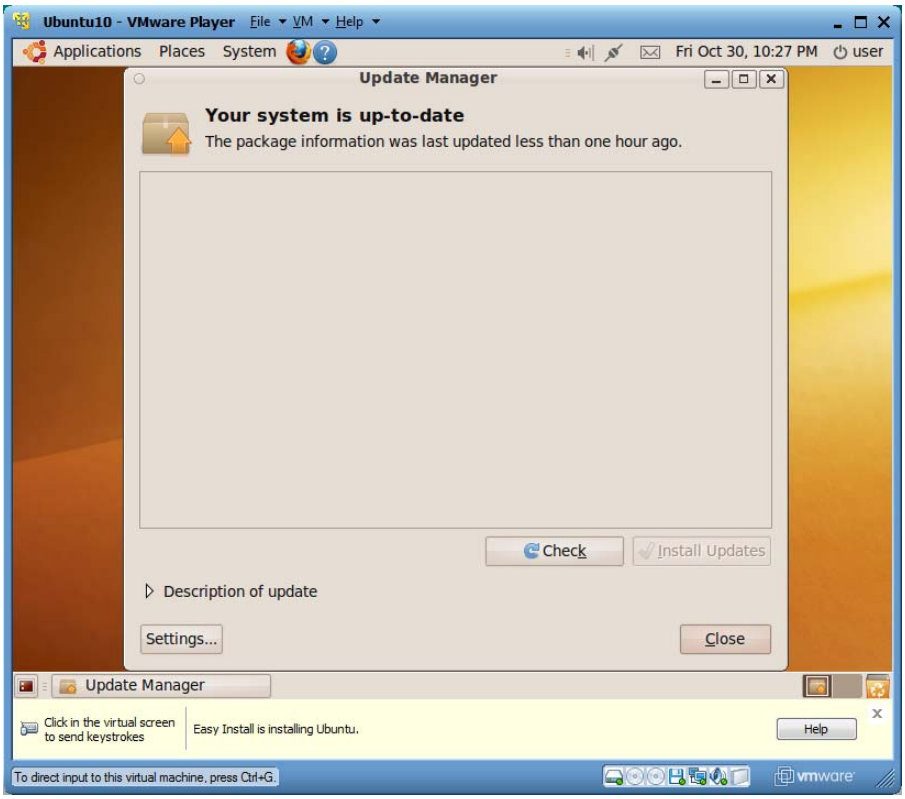

- Click on menu item “System|Administration|Update Manager”.




- Now you see the following “Update Manager” window.

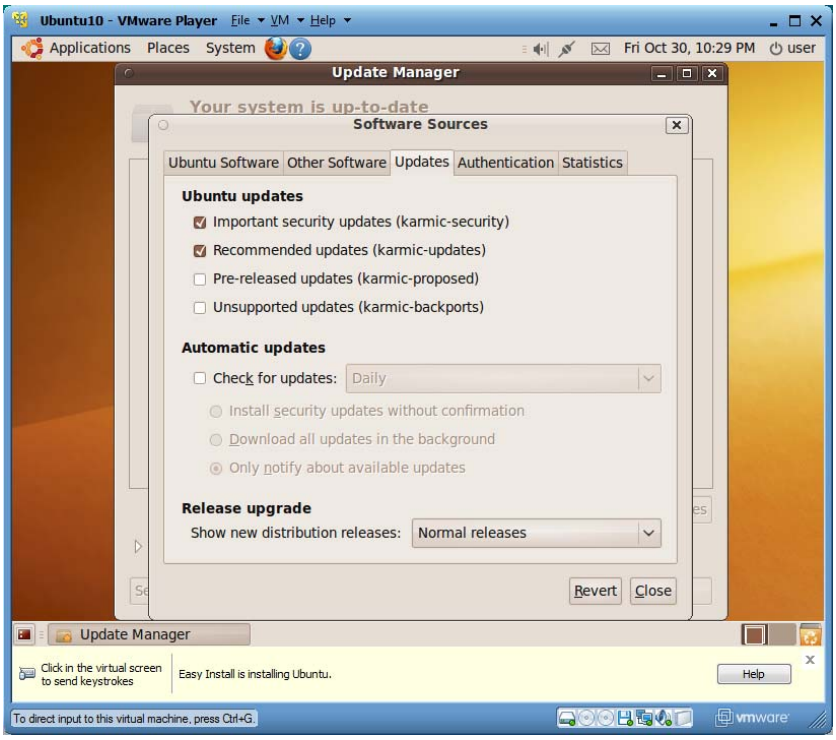



- Click on “Install Updates” to install all available system updates. Wait until the system update is complete, and then scroll down to the bottom of the *Update Manager* window.

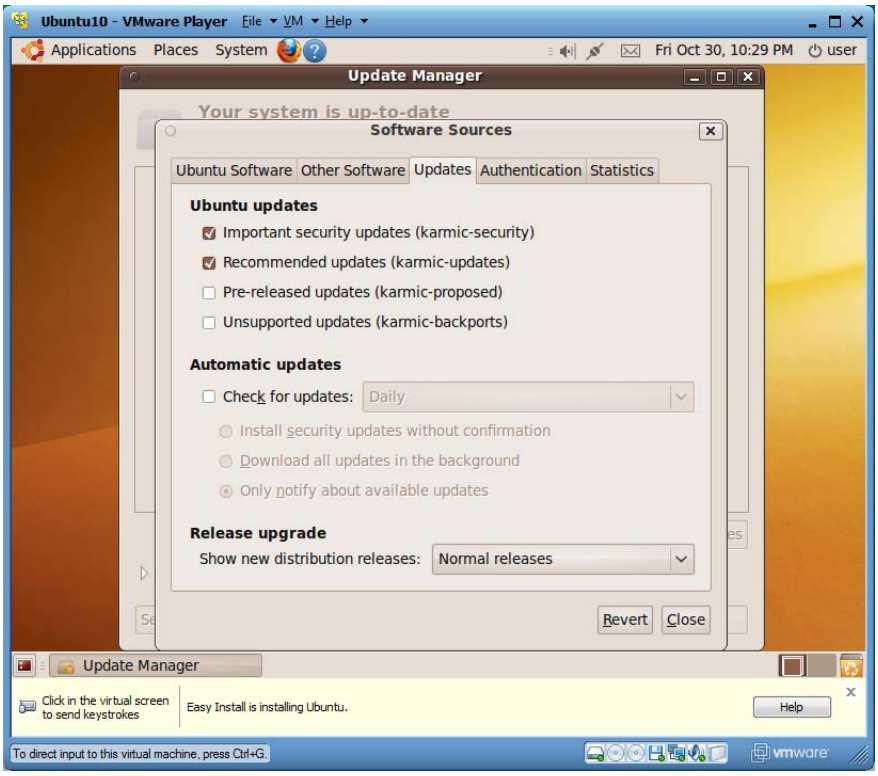



- Click on the “Settings…” button.

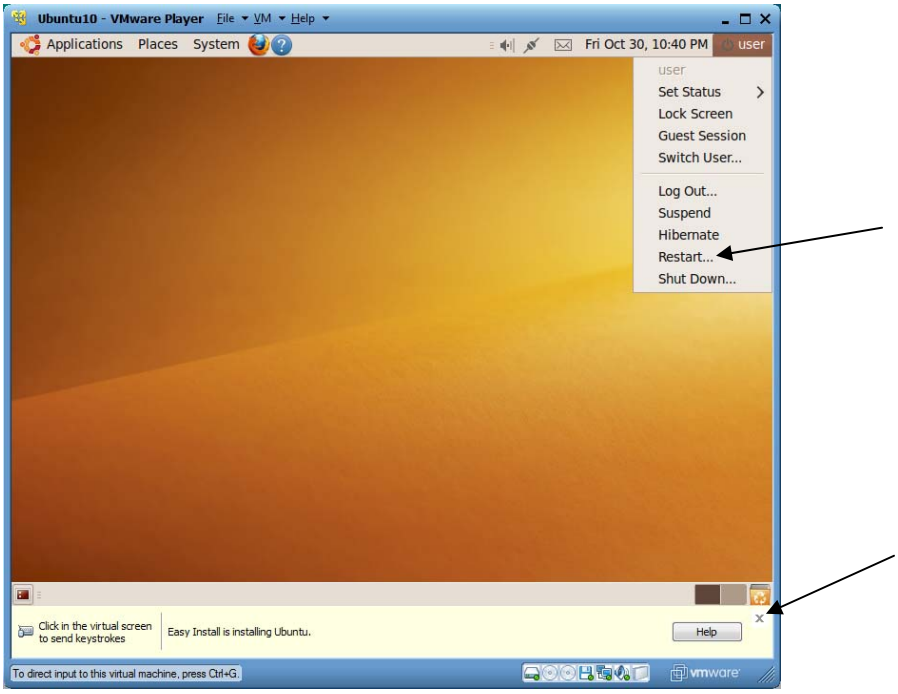
©Copyright 2009 Prof. Lixin Tao



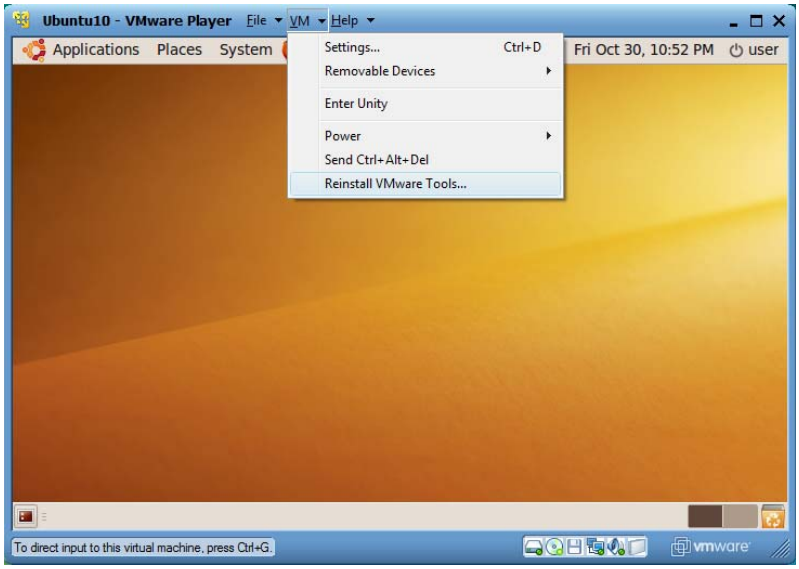

- Uncheck for “Check for updates” so the VM will not always try to download system updates. This is because every time the *Ubuntu* OS gets updated, we may need to reinstall *VMware Tools*, and we try to avoid this trouble.



- Click on the *Close* button to shutdown the "Software Sources" window, and click on the *Close* button again to shutdown the "Update Manager" window.
- Click on the cross x to the bottom-right of the VM window to shut down *Easy Install*. Then click on menu item "user|Restart" to restart your VM.




- After the VM restarts, check menu "VM".

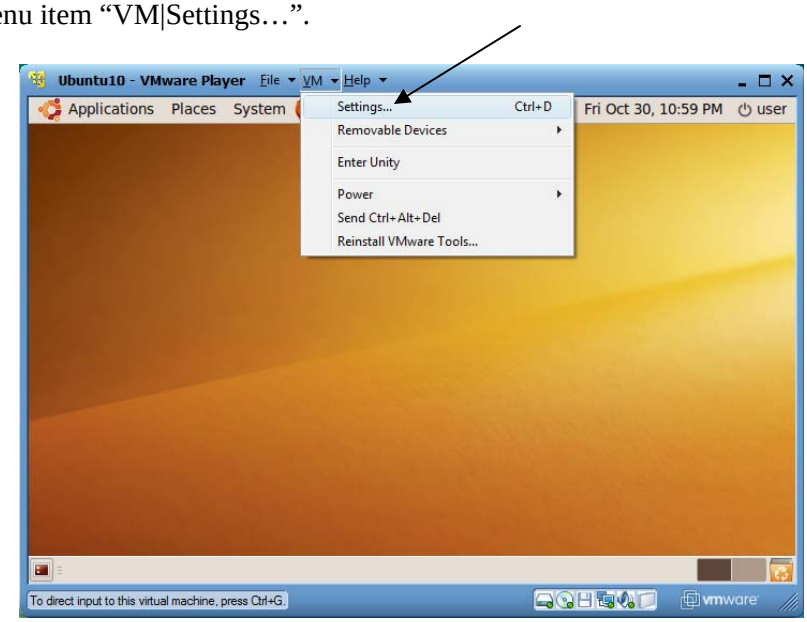



- If the last menu item is "Reinstall VMware Tools…", then *VMware Tools* has been successfully installed and you don't need to reinstall it again unless file drag-and-drop is not working. If the last menu item is "Install VMware Tools…", then choose it to install *VMware Tools*. Most likely *VMware Tools* is installed automatically for you.

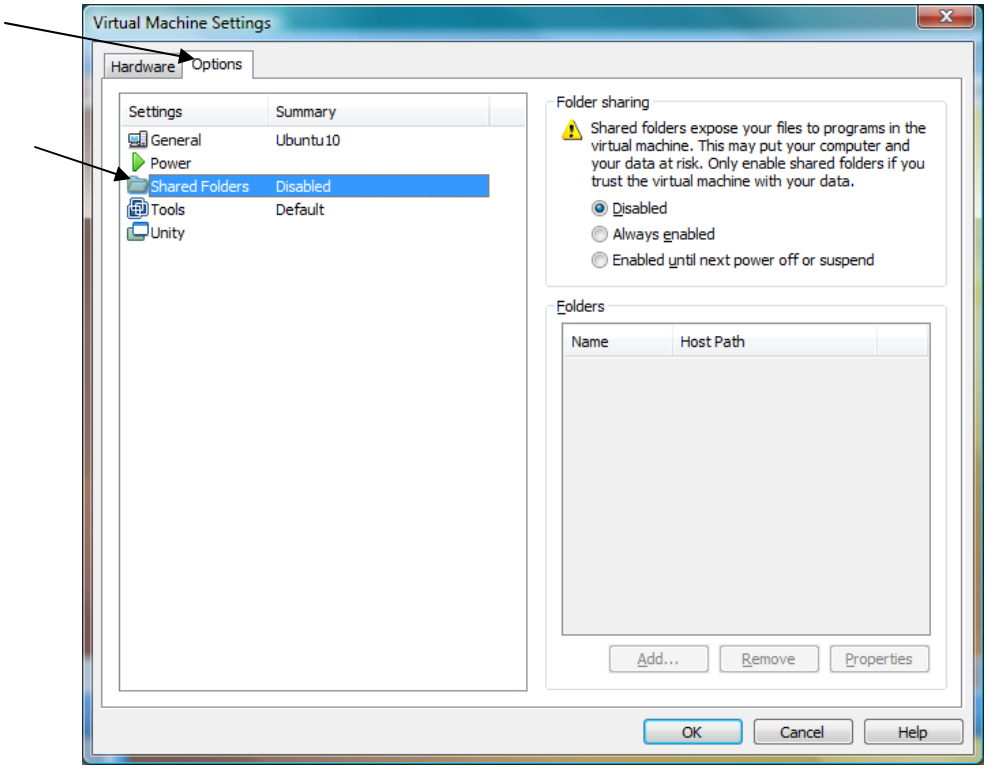
©Copyright 2009 Prof. Lixin Tao

## 3.3 Creating Shared Folder C

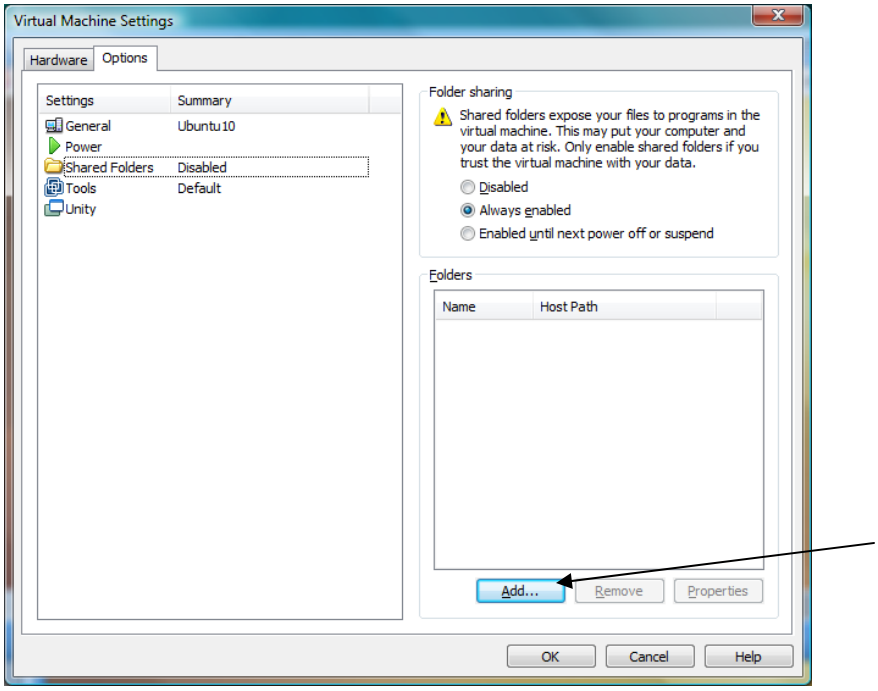
- Click on menu item “VM|Settings…”.




- Click on the “Options” tab of the “Virtual machine Settings”, and then select “Shared Folders”



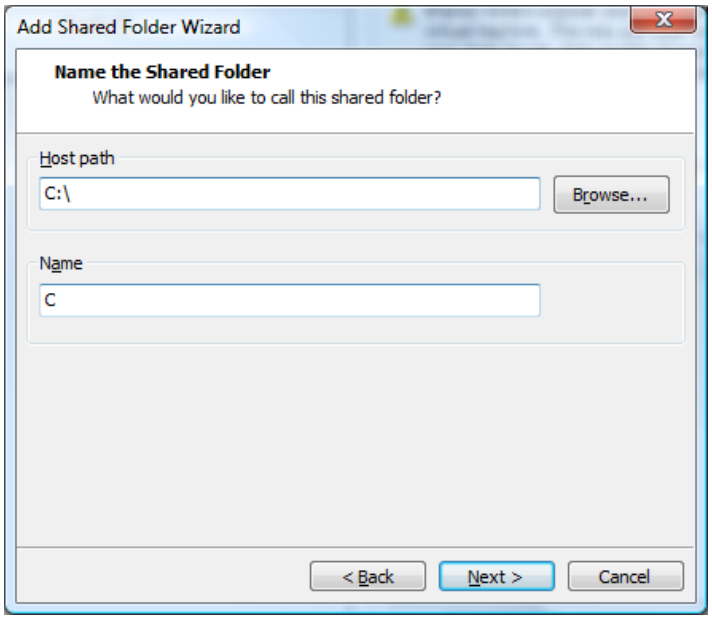
- On the right side, check the “Always enabled” checkbox, and click on the “Add…” button.



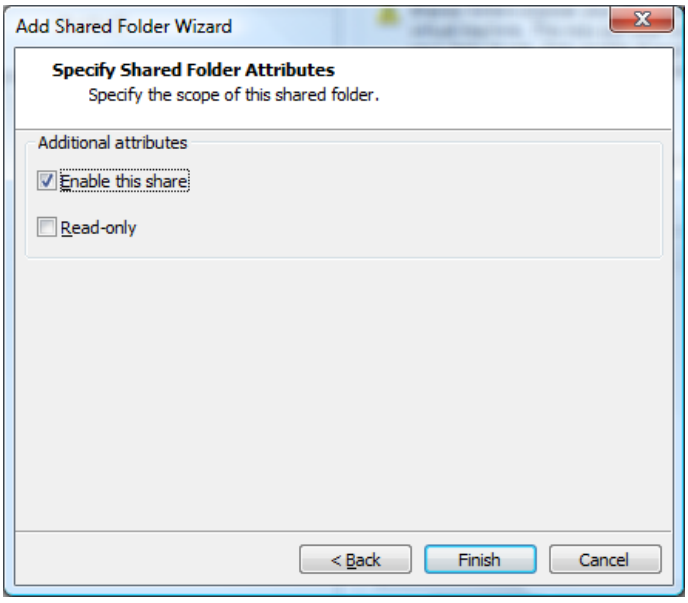

- You will see the following window.

- Click on the *Next* button. Enter “C:\” for *Host Path*, and “C” for *Name*.

©Copyright 2009 Prof. Lixin Tao

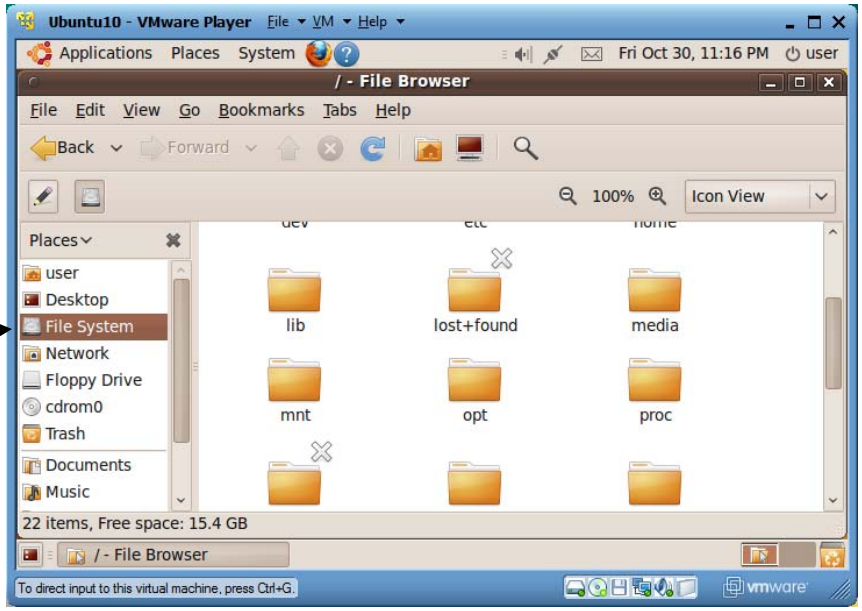



- Click on *the Next* button.



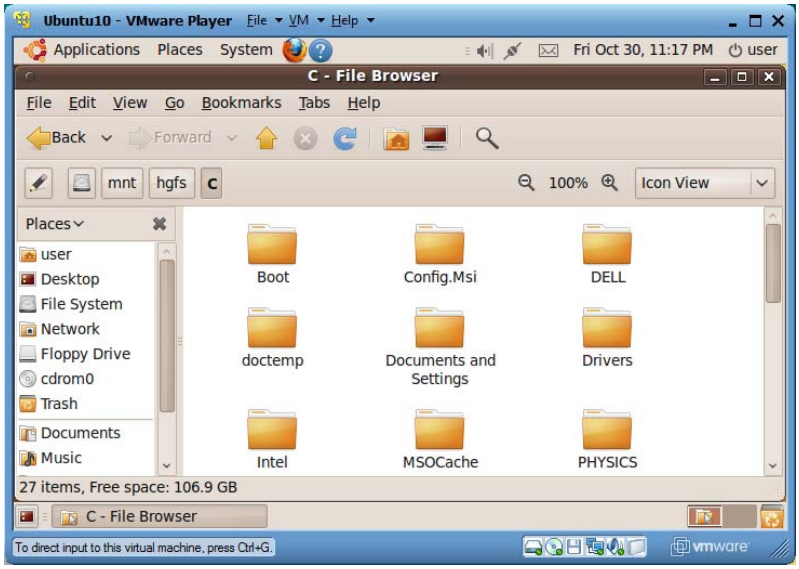

- Make sure "Enable this share" is checked. Click on the *Finish* button to shut down the "Add Shared Folder Wizard".
- Click on the *OK* button to shut down the "Virtual machine Settings" window.
- Now we are ready to check out the shared folder. Click on menu item "Places|Computer".

- Click on “File System” in the left pane.




- Click on *mnt*, then *hgfs*, and then *C* to see all your files and folders on your PC’s drive “C:\”.

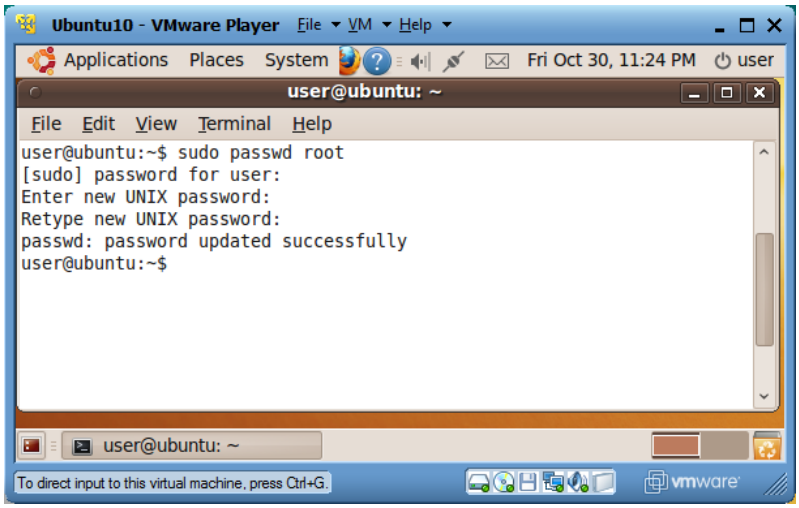
©Copyright 2009 Prof. Lixin Tao




- Congratulations and your shared folder is working.

- For security reason, by default *Ubuntu* doesn't create password for super user *root*. We'd like to create a password for *root*. Click on menu item "Applications|Accessories|Terminal" to launch a terminal window.

- In the terminal window, run "sudo passwd root". When asked for your password, enter 12345678. Then type 12345678 twice as *root*'s password.




Now you have a working Ubuntu V9.10 VM. In the next section you will learn how to install various tools and servers in this VM so it can function as a powerful learning tool for all kinds of information technologies.

©Copyright 2009 Prof. Lixin Tao

# 4
# Installing Applications on a Basic *Ubuntu* Virtual Machine

## 4.1 Overview

This section teaches you how to set up most popular Linux tools and server applications on a basic version of *Ubuntu* V9.10 *VMware* virtual machine (VM), *ubuntu10basic*, which we learned to set up in the last section. The resulting VM will be the same as the completed version downloadable at http://csis.pace.edu/lixin/download/ubuntu10.exe. It is recommended that you first follow my instructions in Section 2 to test drive *ubuntu10* before you try to install the same set of tools and applications yourself.

Even though my instructions are on the installation of applications on a *VMware* virtual machine, they apply to the case when you install the applications on an *Ubuntu*/*Linux* installation on a physical PC. Therefore when we refer to a VM, we also implicitly refer to a physical PC if you are actually doing application installation on a physical Linux PC.

For brevity, when we say to *run "command"*, we mean to start a terminal window (one way is to use *Ubuntu* menu item "Applications|Accessories|Terminal"), and type *command* in that terminal window followed by an *Enter* key, to run the *command*. When we say to *visit "http://url"*, we mean to use a web browser to visit the web page with URL (Uniform Resource Locator) *http://url*. When we say to *click/double-click on a file or folder*, we mean to click/double-click the file or folder in a file explorer. When we say to *click on menu item "A|B"*, we mean clicking on menu "A" to see its popup menu, and then further clicking on menu item "B" on the popup menu.

Since Linux, utility and application versions change all the time, you need to adjust the URLs, installer file names, installation folder names/paths accordingly when necessary. All the instructions are valid on October 31, 2009. The instructions will be revised when necessary.

In *Ubuntu* there are three popular utilities for installing applications: *apt-get*, *aptitude*, and *Synaptic*. Utilities *apt-get* and *aptitude* are both terminal commands and very concise and flexible, while *Synaptic* has a graphic user interface (GUI) but is less flexible. In this guide we mainly use *aptitude* to install applications, which is a wrapper of *apt-get* but with some enhancement. Utility *aptitude* also performs functions that traditionally were handled by many "apt" family utilities. Run "man aptitude" and "man apt-get" to learn more about *aptitude* and *apt-get*. In Linux, if you need to know how to use a command *cmd*, you can try to run "man *cmd*" to read *cmd*'s manual pages, or "*cmd* --help" for a short usage explanation.

While this guide uses *Ubuntu V9.10* as the base Linux system, most of the instructions apply to other flavors of Linux too, including *Red Hat*, *Fedora* (a community version of *Red Hat*) and *Debian*. On *Red Hat* Linux systems, the application installers are normally in the form of RPM format, and utility *yum* is used instead of *aptitude* or *apt-get*.

By now you should have installed *VMware Player*. Otherwise please refer to Section 1 to do so.

## 4.2 Downloading Basic Version of *Ubuntu* V9.10 Linux VM

You can use the VM "C:\VM/ubuntu10" that we created together in Section 3 as the base of application installation in the remainder of this section. If you for reason prefer to skip Section 3 and work directly on this section, you can download the basic VM ubuntu10basic, which is the VM that we should have completed in Section 3, as the foundation VM for this section. To download this completed basic VM, follow the following steps.

- On your PC, create a folder "C:\VM" with *Windows Explorer*.
- Use a web browser to visit http://csis.pace.edu/lixin/download/ubuntu10basic.exe, and save the downloaded file in "C:\VM". The downloaded file "ubuntu10basic.exe" should have 671,788KB.
- In a *Windows Explorer*, double-click on file "C:\VM\ubuntu904basic.exe" to run it, and the execution will generate a folder "C:\VM\ubuntu10basic".
- Rename folder "C:\VM\ubuntu10basic" to "C:\VM\ubuntu10".

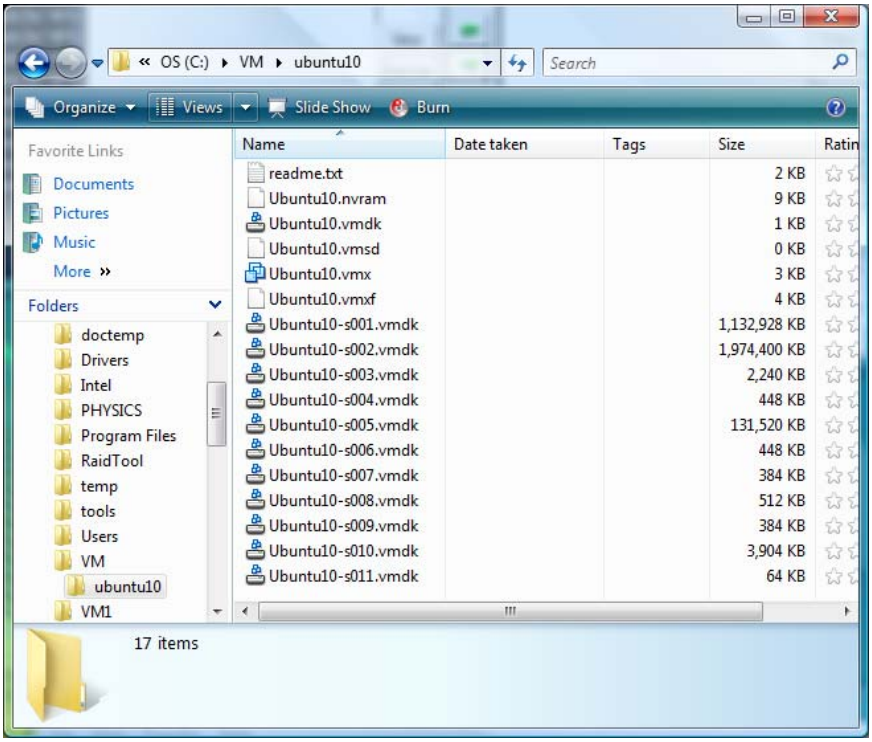
It is highly recommended that you make a copy of this VM so later you could roll back to it if you face some irresolvable problems. You could make a new VM copy (with different folder name) after you complete a major installation in this section so you could easily roll back to the latest good VM version. This section takes over five hours to complete, and the key of success is to follow my instructions to the letters and dots. You can always shut down your VM after any software installation and resume your work at another time.

When you complete all installations in this section, you should have created exactly the same VM as the *ubuntu10* VM that you downloaded from my web site in Section 2.

## 4.3 Launching the VM

In folder "C:\VM\ubuntu10", you will see the following files. If you are using my *ubuntu10basic* VM, file "readme.txt" tells you some information about this VM including the user names and passwords for this VM. There are two pre-set users "root" and "user", both having password "12345678".

©Copyright 2009 Prof. Lixin Tao




Before we start software installation, we use menu item "Applications|Accessories|Terminal" to launch a terminal window, and run the following commands to create a new folder "/home/user/tools" ("~/tools") and "home/user/bin" ("~/bin"). We install major software packages as "user" in its own disk space "~/tools" so we avoid a lot of accessibility problems. When you are more proficient in Linux, you can learn how to install these software as super user *root* and make the software available to all users of your Linux PC. Folder "~/bin" by default is on Linux PATH (a list of folders; Linux will find executable files in these folders in the same order that you list them in PATH) so any executable files on PATH (in one of the folders listed on PATH) are executable by "user" from any working folder.

```
cd
mkdir  tools
mkdir  bin
```

If you copy commands from this guide and paste them into VM terminal window, you need to paste the command in VM terminal window by either terminal window menu item "Edit|Paste" or use key combination "Ctrl+Shift+V". The same key combination can be used to paste text in the *nano* text editor (type "nano" in terminal window to try *nano* text editor). If you are in *gedit* text editor, you can paste by key combination "Ctrl+V".

File "~/.dmrc" is mainly used to remember your Linux user interface language setting. It must be owned by the current user and has 644 as its access code. Folder "/home/user" should not be writable by other users. You may get error messages when you log in if these two conditions are not true. Run the following commands to make sure that "~/.dmrc" and "/home/user" have the correct access rights:

```
chmod  700  /home/user
chmod  644  /home/user/.dmrc
```

We will use Linux package manager *aptitude* to install multiple applications. To make sure your VM's *aptitude* has the most updated information on from which online Linux repository server to download the installers for the applications that you install, run "sudo aptitude update" in a terminal window now. If you are prompted for your password, enter 12345678.

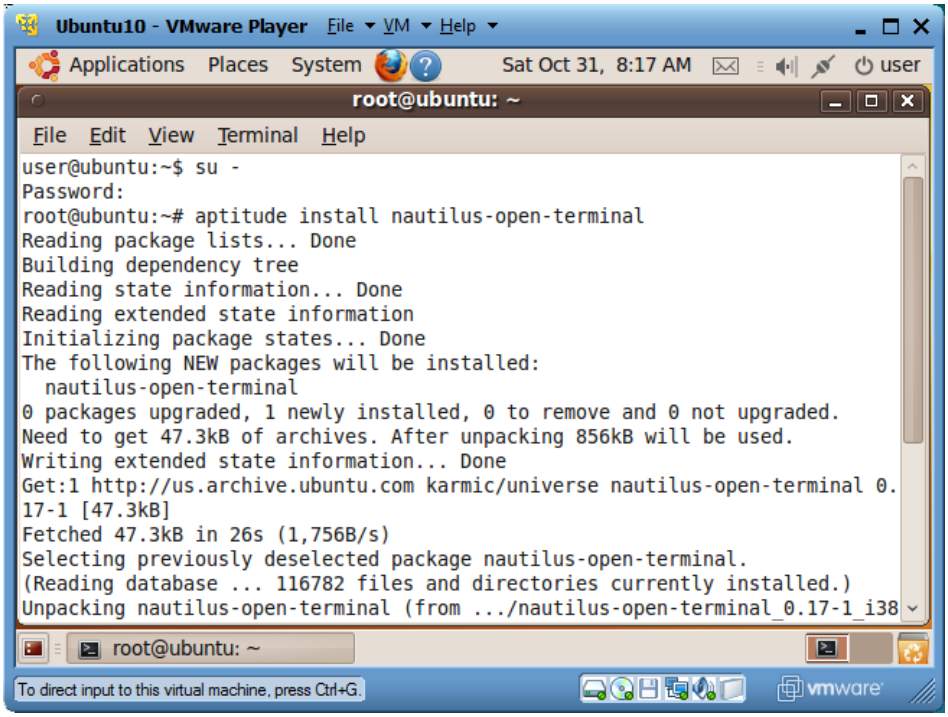
## 4.4 Adding Menu Item "Open In Terminal" to *Nautilus* File Browser Popup Menu

Often you need to open a terminal window, change folder (directory) to a specific folder with command "cd" (*c*hange *d*irectory), and run commands there. If would be nice if we could right-click on a folder in the *Nautilus* file browser, click on an "Open In Terminal" menu item in the popup menu, and have a terminal window launched in that folder. This section shows you how to add such an "Open In Terminal" menu item to the file browser popup menu.

- Start a terminal window by clicking on menu item "Applications|Accessories|Terminal".
- To install software, you normally need be a super user, and "root" is the primary super user. You run command "su -" to request entering super user mode. Upon request, enter *root* password "12345678". Notice that the command prompt changes from $ to #.
- Type

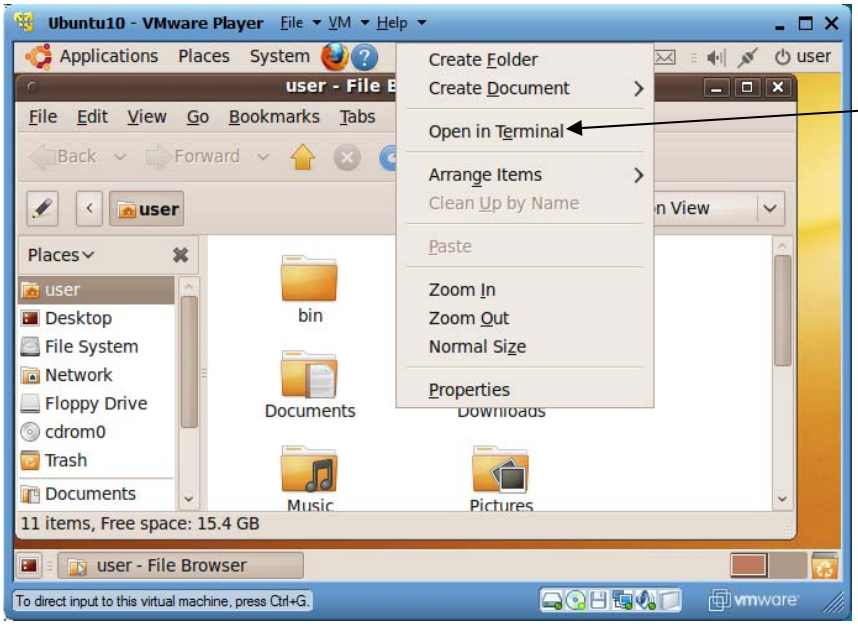
```
aptitude install nautilus-open-terminal
```

  to start the installation.




Reboot the VM to see the new function. In the following example, I opened the filer browser with "Places|Home Folder". The right pane shows folders and files in the home folder "/home/user" of the current user "user". If you right-click on any folder or the blank space in this right pane, you will see a new menu item "Open In Terminal", as shown below. You click on this menu item and a new terminal window will pop up with that folder as the current working folder.

©Copyright 2009 Prof. Lixin Tao




Congratulations and you have installed your first application in Linux!

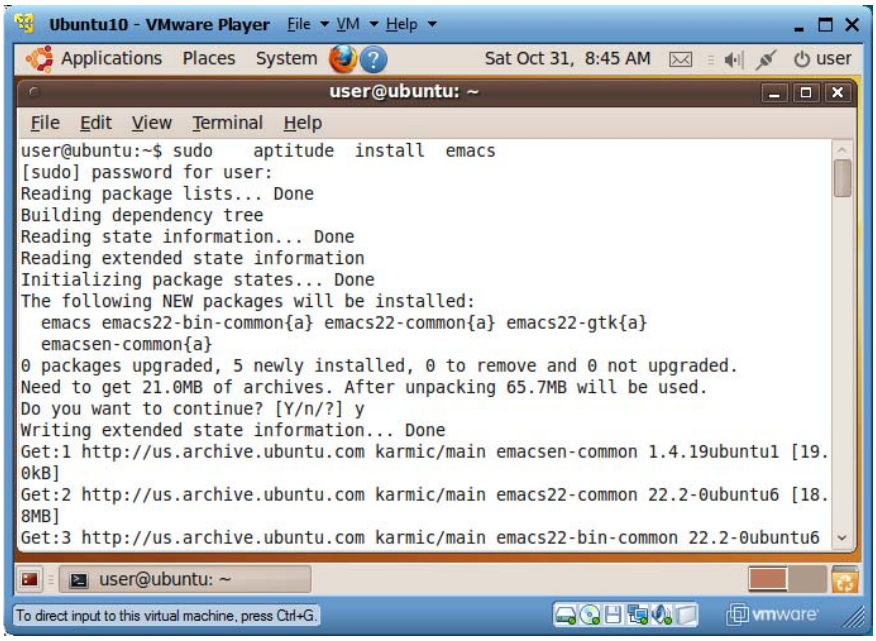
## 4.5 Installing *Emacs* Text Editor

*Ubuntu* comes with three basic text editors: *vi*, *gedit* and *nano*. Editor *vi* is a must for IT staff because it is part of Linux core and thus always available on any Linux installation. But its usage is not intuitive. Here we install a much more powerful text editor *Emacs*, which features multiple editing windows, running programs and keeping long execution logs directly in the editor (if you have experienced "I lost early part of my program output and the terminal window only kept the last section of it", then you will appreciate this feature), and sending emails.

Most software installations should be conducted as super user *root*. In the last section we used "su" to login as root before the software installation. Here I show the second approach: using "sudo" before the actual command to run the command as *root*. Start a new terminal window. In this particular example we install *Emacs* with command

```
sudo  aptitude  install  emacs
```

You will be first prompted for your ("user" instead of "root") password. Type "12345678" as your password.

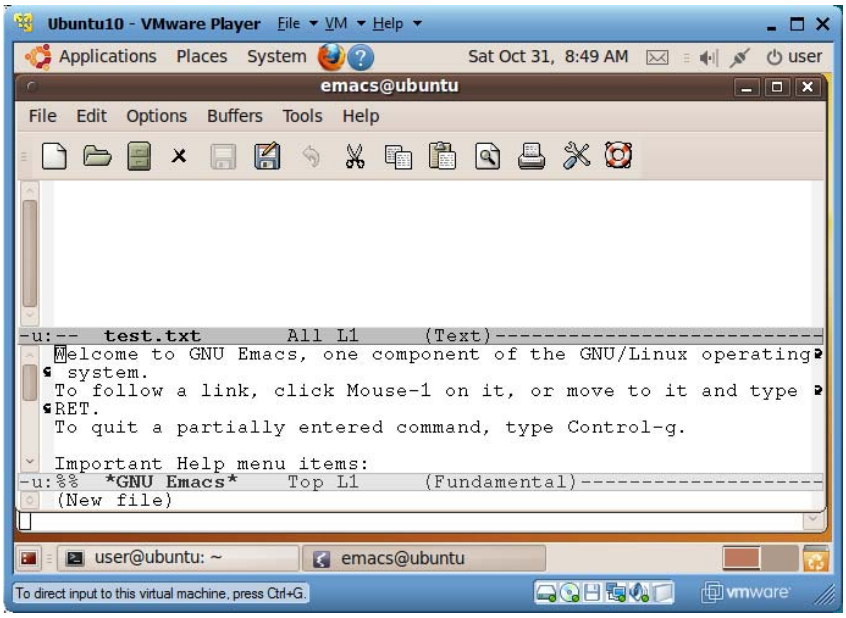



You can use command “locate emacs” to find where are *Emacs* files installed. You can use command “which emacs” to find out which executable is actually executed when you type “emacs” in a terminal window. The following are partial printout of the above two commands.

```
user@ubuntu:~$ locate emacs
/etc/emacs
/etc/emacs/site-start.d
/etc/emacs/site-start.d/50dictionaries-common.el
/etc/speech-dispatcher/clients/emacs.conf
……
user@ubuntu:~$ which emacs
/usr/bin/emacs
```

To launch *Emacs* and edit a new file “test.txt”, type in terminal window command “emacs test.txt”. Use menu item “Help|Emacs Tutorial” to learn how to use *Emacs* later when you really need it. After you type in a terminal window command “emacs test.txt”, that terminal window cannot be used until the editor is shut down. If you want to continue to use the terminal window while the *Emacs* editor is running, launch *Emacs* with commands like “emacs test.txt&”. Ending a command with & is a general way to run the command in background (not being able to interact with the user).

©Copyright 2009 Prof. Lixin Tao




## 4.6 Running Command *sudo* without Needing Password

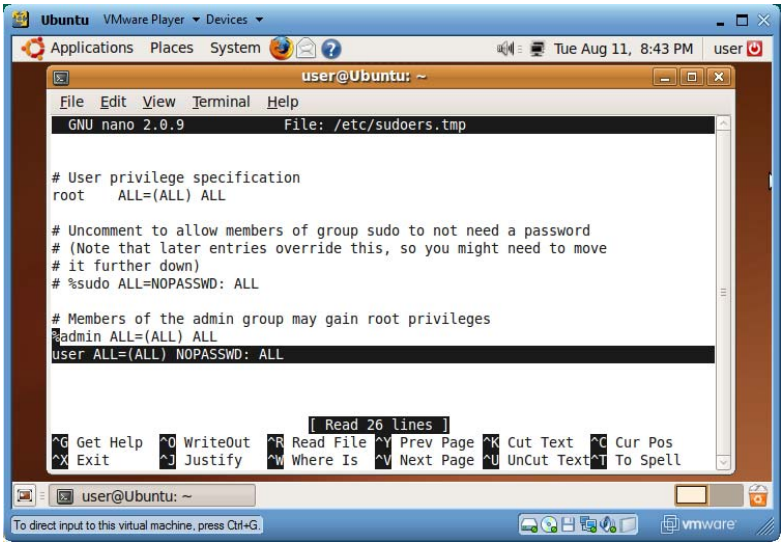
Many programs need be installed or run as super user *root*, but we don't want to share *root*'s password with other people, even if we believe them. Command "sudo" allows a user to run programs as the super user *root*. For example, "sudo ls –alg" may ask you for your Linux password (*sudo* remembers your password for 15 minutes in a terminal window), and then run "ls –alg" as *root*.

File "/etc/sudoers" defines who can assume *root*'s privilege and whether these people need to enter their Linux passwords when they use "sudo". You edit file "/etc/sudoers" by running command "visudo". To allow Linux user "john" to use "sudo" to run commands as root, add line "john ALL=(ALL) ALL" in file "sudoers". When user "john" uses "sudo", he may be prompted to enter his Linux password. If we have instead used line "john ALL=(ALL) NOPASSWD: ALL", then user "john" can use "sudo" without being prompted for his Linux password. You can find more information on "sudo" at http://www.unixtutorial.org/?s=sudoers.

Since we will use *sudo* often to run commands as *root*, we don't want to type our password each time we use *sudo*. We can avoid entering password for using *sudo* by running command "sudo visudo" to use editor *nano* to edit file "/etc/sudoers". Use "Page Down" key to reach the end of the file, and insert the following line:

```
user  ALL=(ALL)  NOPASSWD: ALL
```

Type key combination Ctrl+O and then key *Enter* to write the modified file (/etc/sudoers.tmp) out, and type key combination Ctrl+X to exit the editor (the basic *nano* editing commands are listed at the bottom of the editor window, where ^ means the Ctrl key). This new line in file "/etc/sudoers" specifies that user "user" can run *sudo* without providing his/her password. Now you can test by typing "sudo ls –alg" to run as *root* to list attributes of the files and folders in the current folder, and you will not be prompted to enter your password.




## 4.7 Installing 7z

*7z* and *rar* are popular file zipping/unzipping data formats and algorithms. To get the *7z* and *rar* support in a Linux PC, run "sudo aptitude install p7zip-full". Run command "7z" to see how to use 7z.

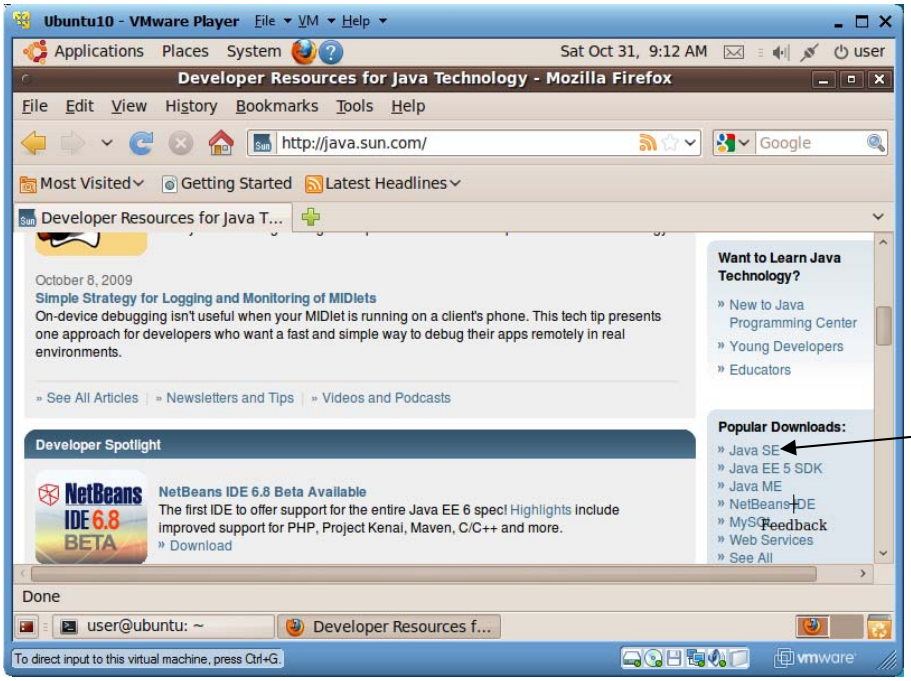
## 4.8 Installing Java JDK, JRE and Java Plugin

Here I show you another software installation method on Linux. You can have more controls of which version of software you want to install, and the installed software files will be centralized in the installation base folder. In this subsection we install the Java JDK (Java Development Kit) package, and Java JRE (Java Runtime Environment) and Java plugin for web browsers are both part of Java JDK.

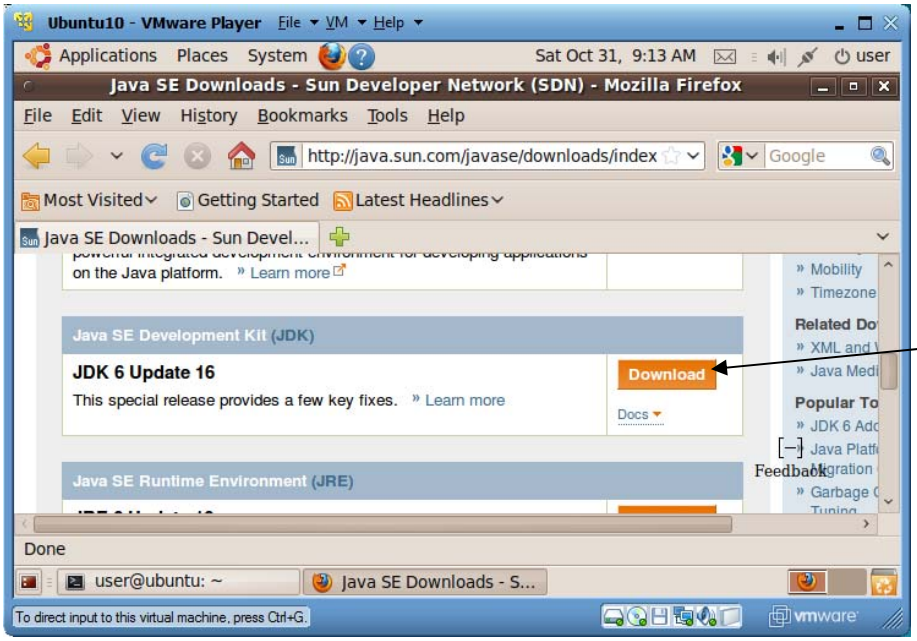
### *4.8.1 Downloading Java JDK*

- Launch VM's *Firefox* web browser and visit *http://java.sun.com*.

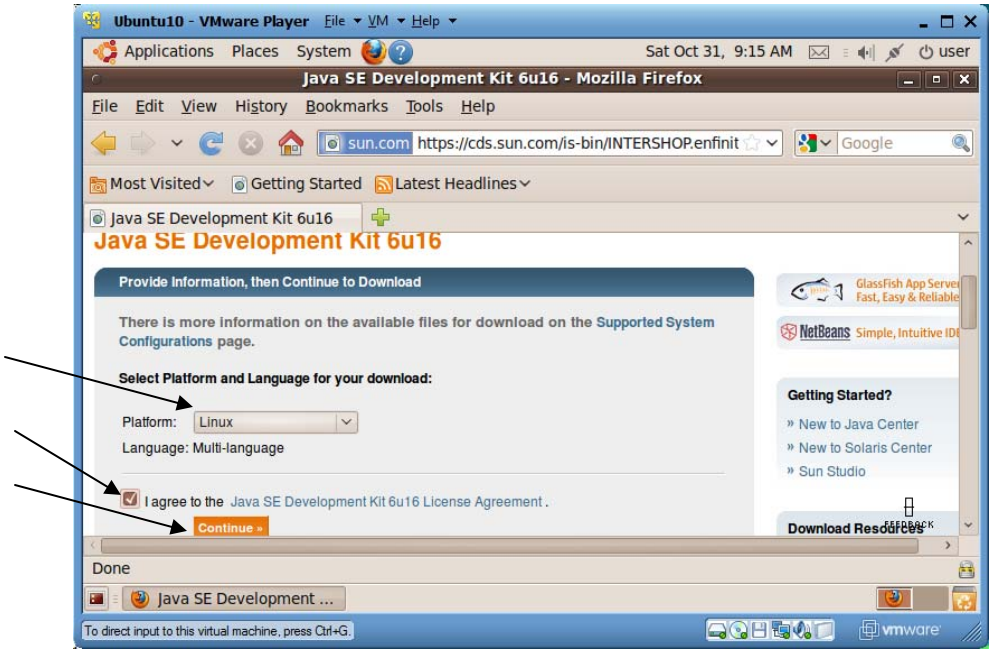
©Copyright 2009 Prof. Lixin Tao



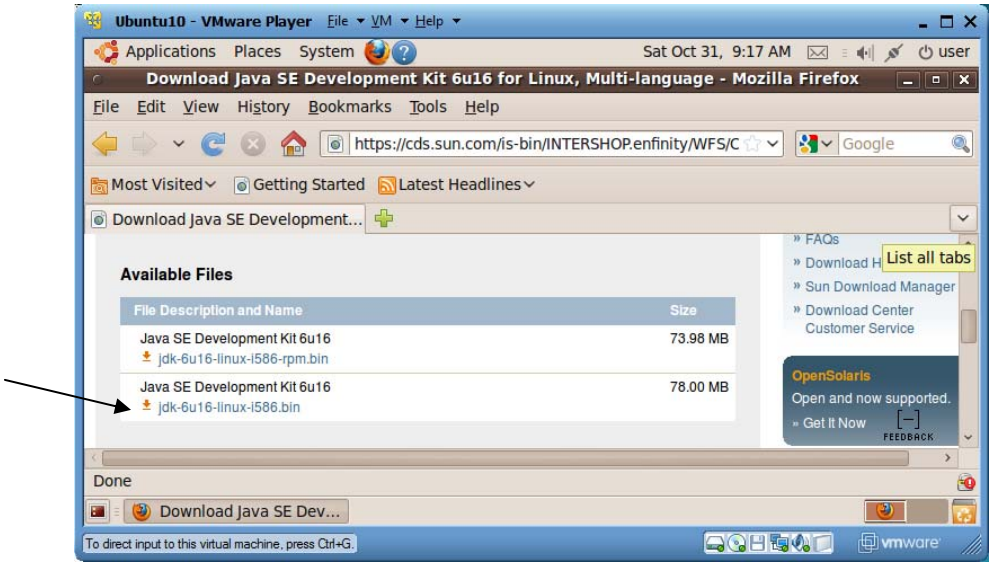

- In the right lower “Popular Downloads” area, click on “Java SE”.




- In the “Java SE Development Kit (JDK) area, click on the *Download* button of the latest Java JDK. For August 2009, the latest version is *JDK 6 Update 16*. The following window shows up.




- Choose “Linux” in the Platform combo box, check the “I agree ….” checkbox, and then click on button *Continue*.




- In the “Available Files” area, click on file “jdk-6u16-linux-i586.bin” (or a similar file) to download it. Make sure the file name should not have “rpm” (rpm version is for *Red Hat* Linux).

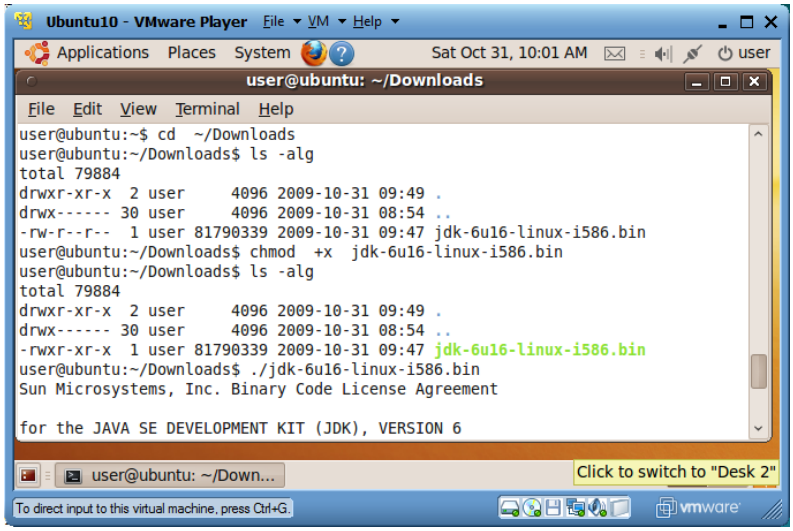
©Copyright 2009 Prof. Lixin Tao

- Choose “Save File”, and click on the *OK* button. The installer has around 78 MB for downloading. The downloaded file “jdk-6u16-linux-i586.bin” will be in folder “/home/user/Downloads”.

### *4.8.2 Installing Java JDK*

When you are logged in as “user”, in a terminal window,

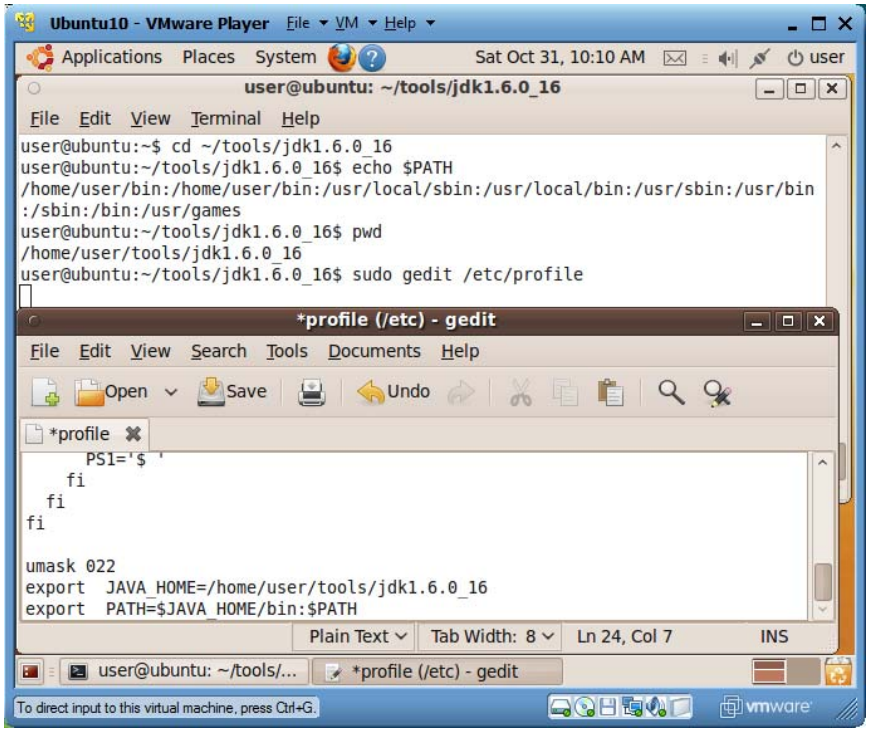
- Run “cd ~/Downloads” to move the working folder to *Desktop*.
- Run “ls –alg” to check out attributes of files or folders in the current folder.
- Run “chmod +x jdk-6u16-linux-i586.bin” to make the file executable.
- Run “ls -alg” to confirm that the file is executable.
- Run “./jdk-6u16-linux-i586.bin”.



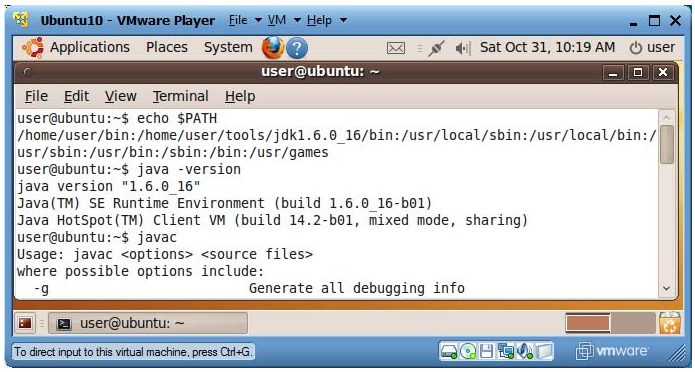

- Use *Space* key to read through the license agreement, and answer Yes to the agreement. Then the installation starts. At the end, type *Space* key and *Enter* key again to complete the installation. You will see that Java JDK has been installed in folder “jdk1.6.0_16” in folder “~/Downloads”.
- Run “mv jdk1.6.0_16 ~/tools” to move the Java JDK installation folder to “/home/user/tools”.
- You could run “rm jdk-6u16-linux-i586.bin” to delete it from folder “~/Downloads”.

- Run “echo $PATH” to see the current definition of environment variable PATH.
- Run “sudo gedit /etc/profile” to add the following two lines to the end of file “/etc/profile”. They define an environment variable JAVA_HOME pointing to the installation folder of Java JDK, and add the “bin” (folder for binary executable commands) of Java JDK on the PATH so any user of this Linux PC can run those commands from any terminal window. Environment variable JAVA_HOME is needed by many Java based applications to find the Java installation on your computer. Every time the Linux PC reboots, the definitions in file “/etc/profile” will be executed/processed.

```
export  JAVA_HOME=/home/user/tools/jdk1.6.0_16
export  PATH=$JAVA_HOME/bin:$PATH
```




- Save the revised file and reboot your Linux PC.
- Run “echo $PATH” to see the revised definition of environment variable PATH.
- Run command “java -version” to see the version of your Java JDK.
- Run command “javac” to see how to use this Java compiler.



©Copyright 2009 Prof. Lixin Tao

### *4.8.3 Installing JRE Plugin for Firefox*

For *Firefox* web browser to support Java applets, we need to install Java JRE Plugin, which is already part of Java JDK installation, for *Firefox*. Run the following commands in a terminal window (if your *Firefox* is a different version you need to adjust folder name "firefox-3.5.4").

```
cd   /usr/lib/firefox-3.5.4/plugins
sudo  ln  -s  ~/tools/jdk1.6.0_16/jre/plugin/i386/ns7/libjavaplugin_oji.so
```

The second command creates a symbolic link, or a shortcut, from "/usr/lib/firefox-3.5.4/plugins/libjavaplugin_oji.so" to "~/tools/jdk1.6.0_16/jre/plugin/i386/ns7/libjavaplugin_oji.so".

## 4.9 Installing *Apache Tomcat*

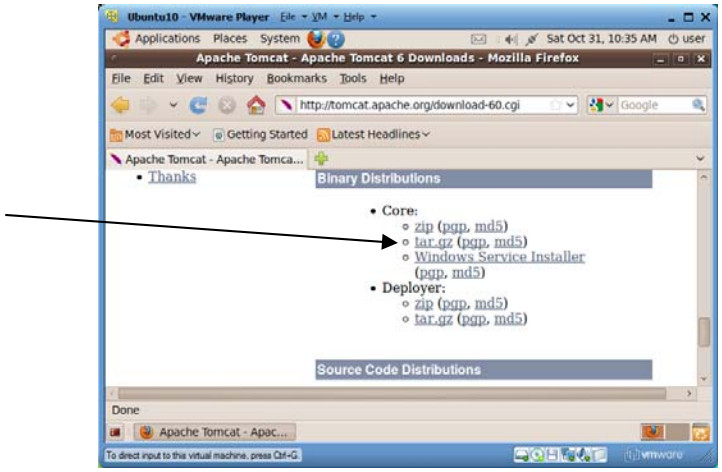
*Apache Tomcat*, later simply called *Tomcat*, is an open-source web server featuring a *Java servlet container* for using Java technologies, including servlets, *JavaServer Pages (JSP)* and *JavaServer Faces (JSF)*, to create HTML files upon client requests over the Internet. It lacks some general web server features like security settings and virtual hosts. On the other hand, *Apache* web server has richer basic web server functions for serving pre-authored (static) HTML files as well as using the older CGI technologies like Perl and PHP to generate HTML files on-the-fly. For a typical enterprise web server, *Apache* web server usually works at front-end serving clients at its default port 80, and if the client request needs be processed by Java technologies, *Apache* would forward (delegate) the request to a *Tomcat* web server behind it. By default *Tomcat* works normally at port 8080.

Before we can install *Tomcat*, we must have installed Java JDK or JRE.

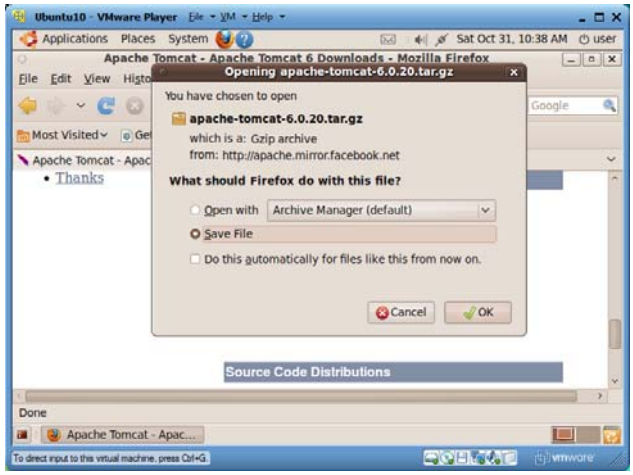
### *4.9.1 Downloading Tomcat Installer*

- Open *Firefox* web browser and visit http://tomcat.apache.org.

- In the left Download area, click on the latest *Tomcat* version. In October 2009, it should be "Tomcat 6.x".




- In the *Core* section of the *Binary Distributions* area, click on "tar.gz" link to download the *Tomcat* installer. In our case the installer is named "apache-tomcat-6.0.20.tar.gz". The following window will pop up.



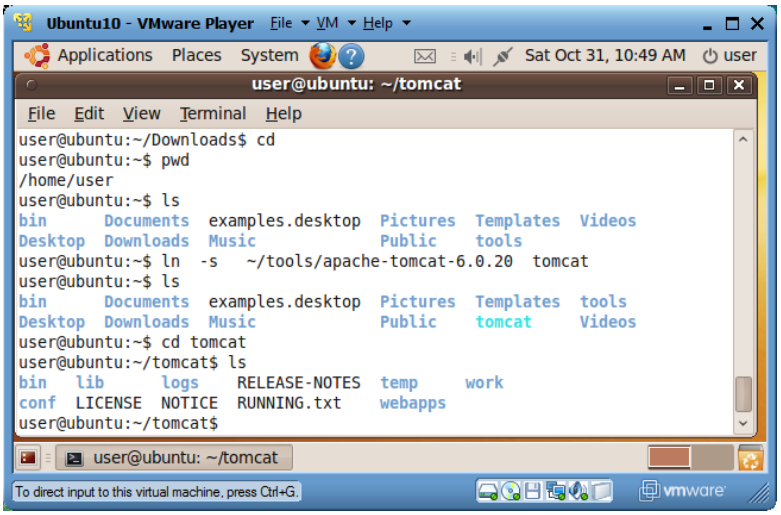

- Check the "Save File" checkbox, and then click on the *OK* button. The installer will be saved in "/home/user/Download". The installer has 5.7MB.

### *4.9.2 Installing Tomcat*

- Run "cd ~/Downloads" and then "ls -alg" to review the downloaded *Tomcat* installer.
- Run "tar xvzf apache-tomcat-6.0.20.tar.gz -C ~/tools" to install *Tomcat* in "/home/user/tools/apache-tomcat-6.0.20".
- You could now run "rm apache-tomcat-6.0.20.tar.gz" to delete the downloaded *Tomcat* installer on your *Desktop*.
- For convenience, we make a symbolic link or shortcut from *user*'s home folder to the *Tomcat* installation folder. We run "cd ~" and "ln -s ~/tools/apache-tomcat-6.0.20 tomcat" to create

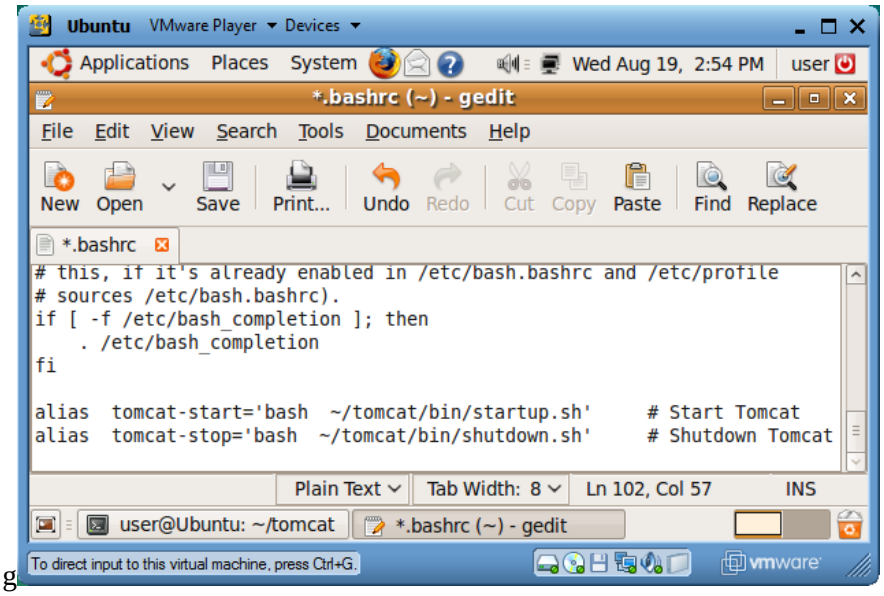
©Copyright 2009 Prof. Lixin Tao

the link "tomcat" in *user*'s home folder. In the following screen capture we also used command "pwd" to show the current folder's path "/home/user", and command "ls" to list files in two folders "~" and "~/tomcat".



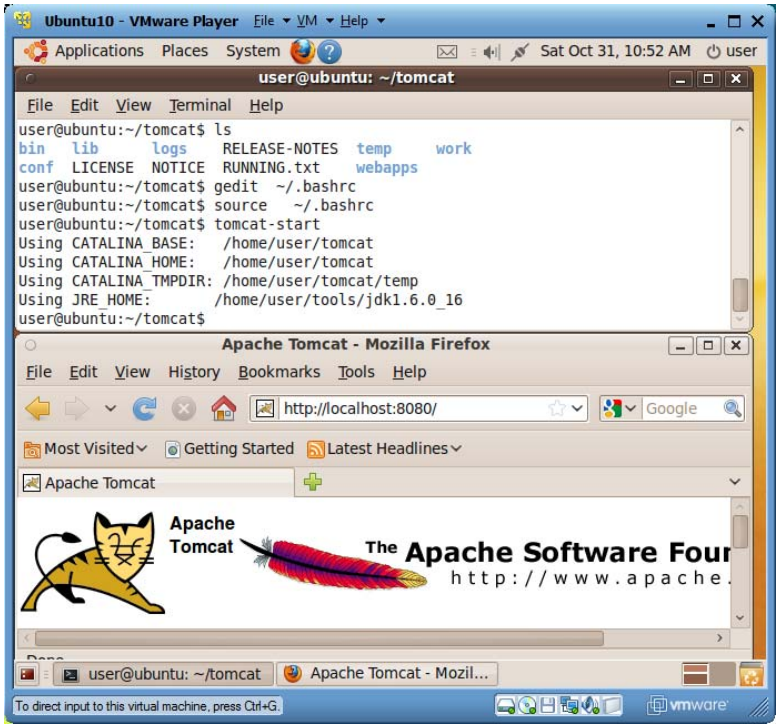

- Run "gedit ~/.bashrc" to use *gedit* to insert the following two lines of alias definitions at the end of file "~/.bashrc" (strings to the right of # are comments):

```
alias  tomcat-start='bash  ~/tomcat/bin/startup.sh'   # Start Tomcat
alias  tomcat-stop='bash  ~/tomcat/bin/shutdown.sh'   # Shutdown Tomcat
```




g

- Run "source ~/.bashrc" or reboot the VM.
- From now on, you can run "tomcat-start" to start the *Tomcat* web server, and run "tomcat-stop" to stop the *Tomcat* web server.
- Run "tomcat-start" to start the *Tomcat* web server. Launch *Firefox* to visit http://localhost:8080. You will see the start page of *Tomcat* as shown below:




- If you need to shut down *Tomcat*, run "tomcat-stop".

### *4.9.3 Tomcat File Organization*

Our *Tomcat* installation base folder is "/home/user/tools/apache-tomcat-6.0.20". The following discussion is relative to this folder.

- All web applications deployed on *Tomcat* are in their own folders under folder "webapps".
- The script files for start/stop *Tomcat* are in folder "bin".
- *Tomcat* configuration files are in folder "conf". File "conf/server.xml" is the main configuration file with which you can change the default port 8080.
- *Tomcat* logging files are in folder "logs". You can read file "logs/catalina.out" to see *Tomcat* startup/stop and error messages. If you print any (debugging) message in your web applications' Java code, the message will be printed to this file. Therefore you can use Java print statements and this file to do simple debugging of your web applications.
- Your web applications' JSP files will be converted to Java servlet source code under folder "work/Catalina/localhost" and organized as web application folders.

### *4.9.4 Changing Tomcat Default Port 8080*

Suppose you need to change *Tomcat*'s default port 8080 to 8090. Run "gedit /home/user/tools/apache-tomcat-6.0.20/conf/server.xml" to edit *Tomcat* configuration file "server.xml". Change the following line

```
<Connector port="8080" protocol="HTTP/1.1"
```

to

©Copyright 2009 Prof. Lixin Tao

```
<Connector port="8090" protocol="HTTP/1.1"
```

Restart *Tomcat* and you can use *Firefox* web browser to visit http://localhost:8090 to verify that *Tomcat* now runs at port 8090. For this tutorial make sure you change *Tomcat* port back to 8080 to be consistent with our later installation guide.

### *4.9.5 Starting Tomcat at Linux Boot-Up Time*

This is an optional step. If you need to start up *Tomcat* automatically at Linux boot up time, you need to install it as a Linux service.

- Run "cd /etc/init.d"
- Run "sudo gedit tomcat" to use *gedit* to create a new script file "tomcat".
- Copy and paste the following contents in file "tomcat" (you may need to adjust the Java installation base folder path or the *Tomcat* installation base folder path; don't change tabs; replacing a tab by space could invalidate the file):

```
# Tomcat auto-start
#
# description: Auto-starts tomcat
# processname: tomcat

export JAVA_HOME=/home/user/tools/jdk1.6.0_16
case $1 in
start)
        sh  /home/user/tools/apache-tomcat-6.0.20/bin/startup.sh
        ;;
stop)
        sh  /home/user/tools/apache-tomcat-6.0.20/bin/shutdown.sh
        ;;
restart)
        sh  /home/user/tools/apache-tomcat-6.0.20/bin/shutdown.sh
        sh  /home/user/tools/apache-tomcat-6.0.20/bin/startup.sh
        ;;
esac
exit 0
```

- Run "sudo chmod 755 tomcat" to make script file "tomcat" executable.
- Run the following two commands to create symbolic links to script file "tomcat" in "/etc/rc1.d" and "/etc/rc2.d":

```
sudo ln -s /etc/init.d/tomcat /etc/rc1.d/K99tomcat
sudo ln -s /etc/init.d/tomcat /etc/rc2.d/S99tomcat
```

When Linux boots up, all scripts in "rc2.d" with "S" as name initial are executed to start some services. Therefore *Tomcat* gets launched when Linux starts. When Linux shuts down and just loses its network access (run level 1), all scripts in folder "rc1.d" with "K" as name initial are executed to shut down some services. Therefore *Tomcat* shuts down immediately before the Linux halts. You can learn more about Linux run level at

http://www.networkclue.com/os/Linux/run-levels.aspx. You can learn more about Linux startup process at http://floppix.ccai.com/rc.html.

- Reboot the VM, and launch *Firefox* web browser to visit http://localhost:8080. You should see that *Tomcat* is running.
- Later, in case you don't want *Tomcat* to start at boot-up time, just remove the two links "/etc/rc1.d/K99tomcat" and "/etc/rc2.d/S99tomcat".

## 4.10 Installing *Apache* Web Server and Its CGI Plugins

### *4.10.1 Installing Apache 2*

*Apache* web server is the most popular open-source web server. This subsection explains how to install it.

- Run "sudo aptitude install apache2" to download and install *Apache*. If you see 404 error messages, make sure you update your VM's *aptitude* information by running "sudo aptitude update" and then try this step again.
- Run "sudo gedit /etc/apache2/httpd.conf" and insert the following line in it:

```
ServerName    localhost
```

- Run command "ln -s /var/www /home/user/www" to create a symbolic link or shortcut "/home/user/www" to "/var/www", the document root folder for *Apache*. Each web site of yours on *Apache* will be in a folder under "/var/www". The *Apache* web server's installation folder is "/etc/apache2".

```
user@ubuntu:~$ sudo aptitude install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
The following NEW packages will be installed:
  apache2 apache2-mpm-worker{a} apache2-utils{a} apache2.2-bin{a}
  apache2.2-common{a}   libapr1{a}   libaprutil1{a}   libaprutil1-dbd-
sqlite3{a}
  libaprutil1-ldap{a}
0 packages upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 2,088kB of archives. After unpacking 6,939kB will be used.
Do you want to continue? [Y/n/?] y
……
user@ubuntu:~$ sudo gedit /etc/apache2/httpd.conf
user@ubuntu:~$ ln -s /var/www /home/user/www
user@ubuntu:~$ ls
bin      Documents  examples.desktop  Pictures  Templates  tools   www
Desktop  Downloads  Music             Public    tomcat     Videos
user@ubuntu:~$ cd www
user@ubuntu:~/www$ ls
index.html
user@ubuntu:~/www$ more index.html
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
```

©Copyright 2009 Prof. Lixin Tao

```
   <p>The web server software is running but no content has been added,
yet.</p>
   </body></html>
   user@ubuntu:~/www$ cd /etc/apache2
   user@ubuntu:/etc/apache2$ ls
   apache2.conf  httpd.conf   mods-available  sites-available
   conf.d        httpd.conf~  mods-enabled    sites-enabled
   envvars       magic        ports.conf
   user@ubuntu:/etc/apache2$
```

- Launch *Firefox* web browser and visit http://localhost. You will see a screen similar to the following one. *Firefox* is rendering HTML file "/var/www/index.html". *Apache* has been installed as a Linux service and it will automatically start at Linux boot-up time.

- At any time, you can restart *Apache* web server by running "sudo apache2ctl restart".
- Another way to restart *Apache* is to run "sudo /etc/init.d/apache2 restart". For convenience, run "gedit ~/.bashrc" and insert the following line at the end of the file:

  alias apache='sudo /etc/init.d/apache2'

- Run "source ~/.bashrc". From now on you can also use "apache restart" to restart *Apache*.

### *4.10.2 Installing PHP*

*PHP* is a CGI (Common Gateway Interface) technique for dynamically generating HTML files based on clients' HTTP requests. It is more powerful than *Perl*. To install *PHP* v5 on *Apache*, run the following commands:

- Run "sudo aptitude install php5 libapache2-mod-php5" to install *PHP5* on your Linux system.

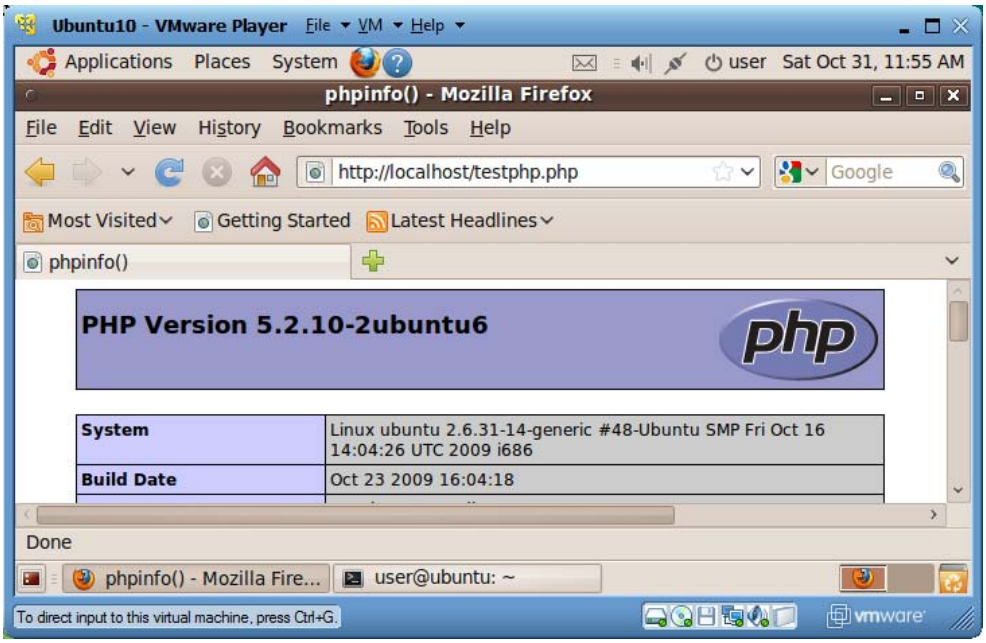
```
   user@ubuntu:~$ sudo  aptitude  install  php5  libapache2-mod-php5
   Reading package lists... Done
   Building dependency tree
   Reading state information... Done
```

```
Reading extended state information
Initializing package states... Done
The following NEW packages will be installed:
  apache2-mpm-prefork{a} libapache2-mod-php5 php5 php5-common{a}
The following packages will be REMOVED:
  apache2-mpm-worker{a}
0 packages upgraded, 4 newly installed, 1 to remove and 0 not upgraded.
Need to get 2,929kB of archives. After unpacking 6,304kB will be used.
Do you want to continue? [Y/n/?] y
……
user@ubuntu:~$ sudo a2enmod php5
Module php5 already enabled
user@ubuntu:~$ sudo apache2ctl restart
```

- Run command "sudo apache2ctl restart" to restart *Apache* web server.
- To test your *PHP5* installation, run command "sudo gedit ~/www/testphp.php", and insert the following contents into the file:

```
<?php  phpinfo(); ?>
```

- Save the file, and use *Firefox* to visit http://localhost/testphp.php. If you see a screen similar to the following one, you have succeeded in installing *PHP5* in *Apache*.




- To find out what other *PHP5* modules that you can install with "sudo aptitude install *php5-module*", run command "aptitude search php5".

### *4.10.3 Installing Perl*

*Perl* is a more primitive form of scripting language used for earlier CGI support. It is mainly used by technicians and students to write simple test web applications. The following steps show how to set up *Perl* support in *Apache*.

©Copyright 2009 Prof. Lixin Tao

- Run "sudo aptitude install libapache2-mod-perl2" to download and install *Perl* for *Apache*.
- Run "sudo mkdir ~/www/cgi-bin" to create a folder "cgi-bin" under *Apache* document root folder for holding *Perl* scripts.
- Run command "sudo gedit /etc/apache2/sites-enabled/000-default" and update file "000-default"'s contents section containing "/cgi-bin/" so it reads as below (we actually only need to update the path of "cgi-bin" in the first two lines, and insert the 7th line below). This update tells *Apache* that if a client HTTP request URL contains substring "/cgi-bin/file-name" where the "file-name" has file name extension ".cgi" or ".pl", *Apache* would run script "/var/www/cgi-bin/file-name" to process this HTTP request.

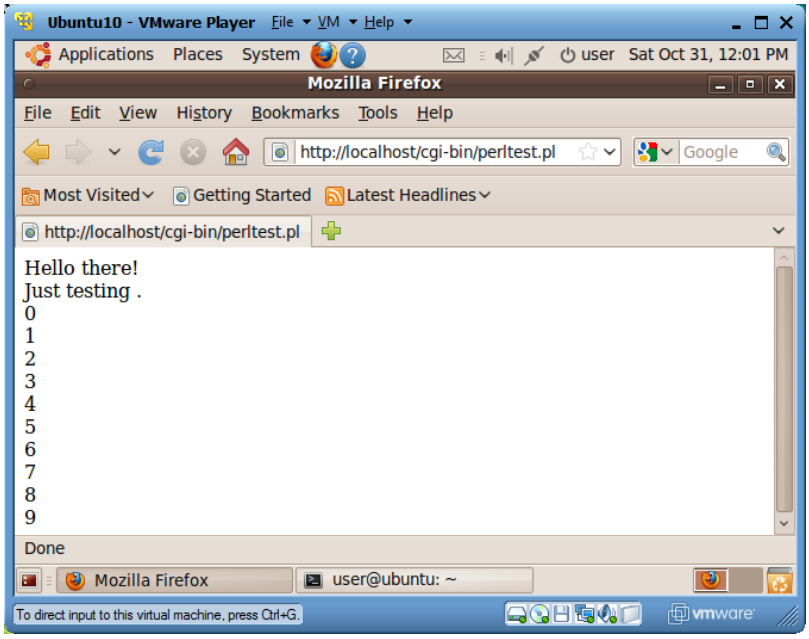
```
ScriptAlias /cgi-bin/ /var/www/cgi-bin/
<Directory "/var/www/cgi-bin">
      AllowOverride None
      Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
      Order allow,deny
      Allow from all
      AddHandler cgi-script cgi pl
</Directory>
```

- Run command "sudo apache2ctl restart" to restart *Apache*.
- To test your *Perl* installation, run command "sudo gedit ~/www/cgi-bin/perltest.pl" to create a new file "perltest.pl" in "/home/user/www/cgi-bin", and insert the following contents in this file:

```
#!/usr/bin/perl -w
print "Content-type: text/html\r\n\r\n";
print "Hello there!<br />\nJust testing .<br />\n";

for ($i=0; $i<10; $i++)
{
    print $i."<br />";
}
```

- Save the file, and run "sudo chmod a+x ~/www/cgi-bin/perltest.pl" to make the file executable by all users.
- Now open *Firefox* web browser at http://localhost/cgi-bin/perltest.pl, and you will see a screen similar to the following one:




### *4.10.4 Integrating Apache with Tomcat*

Before working on this subsection, you must have installed *Tomcat*.

The following steps configure *Apache* and *Tomcat* so *Apache* will delegate a chosen set of web applications for *Tomcat* to process. As a result the remote users can access these *Tomcat* web applications without using port 8080.

- Use a text editor to check whether file "~/tools/apache-tomcat-6.0.20/conf/server.xml" has a line similar to the following one. If it has, then *Tomcat* is ready and we don't need to do anything for it (the new *Tomcat* releases are all ready). Otherwise insert such a line nested inside the "<Service name="Catalina">" element. Our *Tomcat* installation already has a similar line "<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />" so we don't need to do anything here. This line enables AJP connections from *Apache* (or another web server) to port 8009 of your *Tomcat* server.

```
    <Connector port="8009" enableLookups="false" redirectPort="8443"
               protocol="AJP/1.3" />
```

- Run "sudo gedit /etc/apache2/sites-enabled/000-default" to add in file "000-default" the following lines just before </VirtualHost> for the default *Tomcat* web applications that you need delegate by *Apache* (for each of your *Tomcat* web application that you need *Apache* to delegate, add a similar pair of *ProxyPass* and *ProxyPassReverse* declarations):

```
    ProxyPass /examples/servlets ajp://localhost:8009/examples/servlets
    ProxyPassReverse /examples/servlets ajp://localhost:8009/examples/servlets

    ProxyPass /examples/jsp ajp://localhost:8009/examples/jsp
    ProxyPassReverse /examples/jsp ajp://localhost:8009/examples/jsp
```

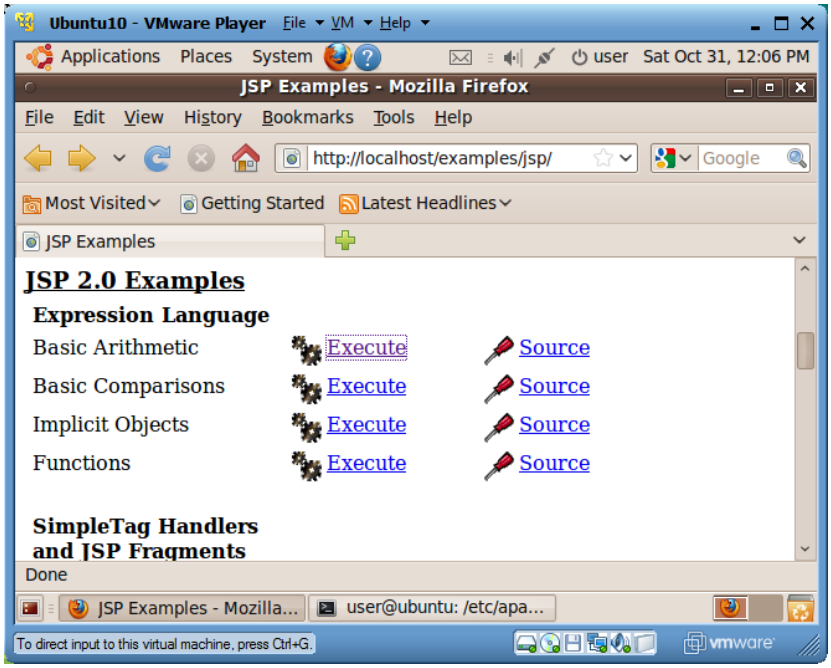
©Copyright 2009 Prof. Lixin Tao

```
ProxyPass /docs ajp://localhost:8009/docs
ProxyPassReverse /docs ajp://localhost:8009/docs
```

- Run command "cd /etc/apache2/mods-available" to change working folder to the folder holding all available *Apache* modules.
- Run command "sudo a2enmod proxy" and "sudo a2enmod proxy_ajp" to enable *Apache* modules *proxy* and *proxy_ajp*.
- We need to disable file "/etc/apache2/mods-enabled/proxy.conf". Run the following commands:

```
cd /etc/apache2/mods-enabled
sudo mv proxy.conf proxy.conf.disable
```

- Restart *Apache* by running command "sudo apache2ctl restart" (don't use other ways of restarting *Apache*).
- To test your configuration, launch *Firefox* to visit http://localhost:8080/examples/jsp. If the page doesn't show up, run "tomcat-start" to start *Tomcat*. Otherwise remove the port number 8080 and observe that http://localhost/example/jsp also shows the same page, as shown below. This confirms that *Apache* is delegating the user request to *Tomcat*.




## 4.11 Installing *MySQL* Database Server

*MySQL* community version is a popular open-source database server from *Oracle*. This guide shows how to install it on our VM.

- Run command "sudo aptitude install mysql-server" to download and install the current *MySQL* server prepared for Linux. At the time of writing it installs MySQL 5.1.

- When asked for password for *root* (*MySQL* account *root*; it is different from Linux's super user *root*), enter 123456 (make sure you use this password because some of our example web applications use it; you can change the password later after you are proficient in Linux).
- *MySQL* will be installed as a Linux service, and it will start automatically at system boot up time.
- Many of our example web applications will use a database named "test". For security reason our installer doesn't create this "test" database for us. We now create it with *MySQL*'s administrator console.
    - Run command "mysql -u root -p123456" to login *MySQL* admin console with *root* as user name and 123456 as password.
    - After the *MySQL* command prompt "mysql>", use command "create database test;" to create database "test", use command "show databases;" to view a list of available databases, use command "use test;" to use database "test" as the current one so the following commands apply to it, use "show tables;" to list the tables in the current database, and use "quit;" to quit the *MySQL* admin console.
    - If you have a file of SQL statements named, say "survey.sql", in the current working folder, when you are still in *MySQL* admin console, you can use "source survey.sql;" to run the SQL statements in file "survey.sql" and apply them to the current database.

  The following screen capture shows these steps' execution (we will show "source" example when we deploy example web applications in *Tomcat*).

```
user@ubuntu:~$ sudo  aptitude  install  mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
The following NEW packages will be installed:
  libdbd-mysql-perl{a} libdbi-perl{a} libhtml-template-perl{a}
  libnet-daemon-perl{a} libplrpc-perl{a} mysql-client-5.1{a}
  mysql-server mysql-server-5.1{a} mysql-server-core-5.1{a}
0 packages upgraded, 9 newly installed, 0 to remove and 0 not
upgraded.
Need to get 20.4MB of archives. After unpacking 48.8MB will be used.
Do you want to continue? [Y/n/?] y
Writing extended state information... Done
……
user@ubuntu:~$ mysql  -u root  -p123456
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 40
Server version: 5.1.37-1ubuntu5 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.
Mysql> create  database  test;
Query OK, 1 row affected (0.03 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
```

©Copyright 2009 Prof. Lixin Tao

```
| mysql              |
| test               |
+--------------------+
3 rows in set (0.00 sec)

mysql> use test;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> quit;
Bye
user@ubuntu:~$
```

## 4.12 Installing *NetBeans and Sun's Java EE Bundle*

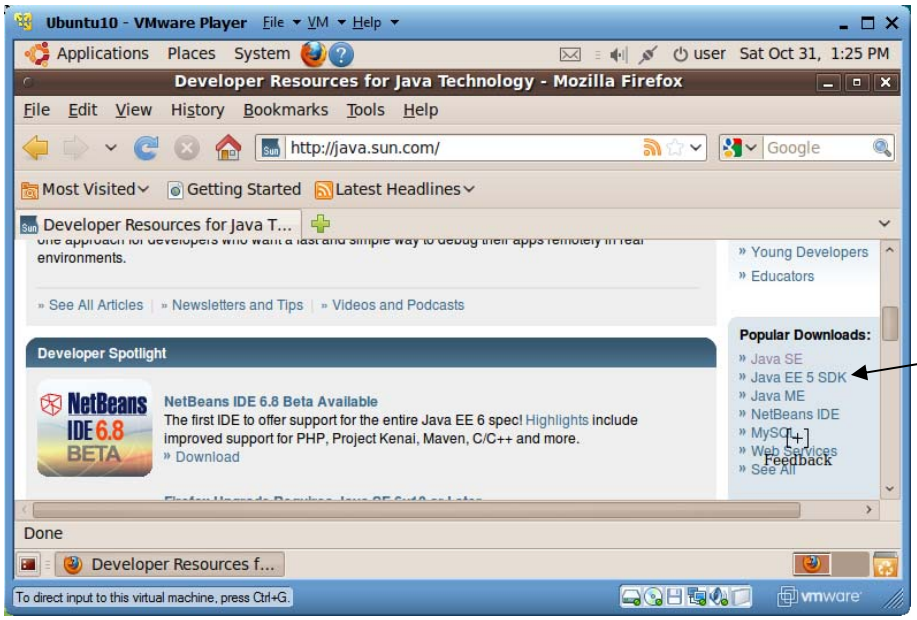
Before we can install *NetBeans*, we must have installed Java JDK.

*NetBeans* is a popular open-source Java IDE that can be used to develop software projects in multiple languages. *NetBeans* is from *Sun Microsystems*, and it features a quality What-You-See-What-You-Get (WYSWYG) visual graphic user interface designer. Since *NetBeans* is also implemented in Java, you must install Java JDK or JRE first.

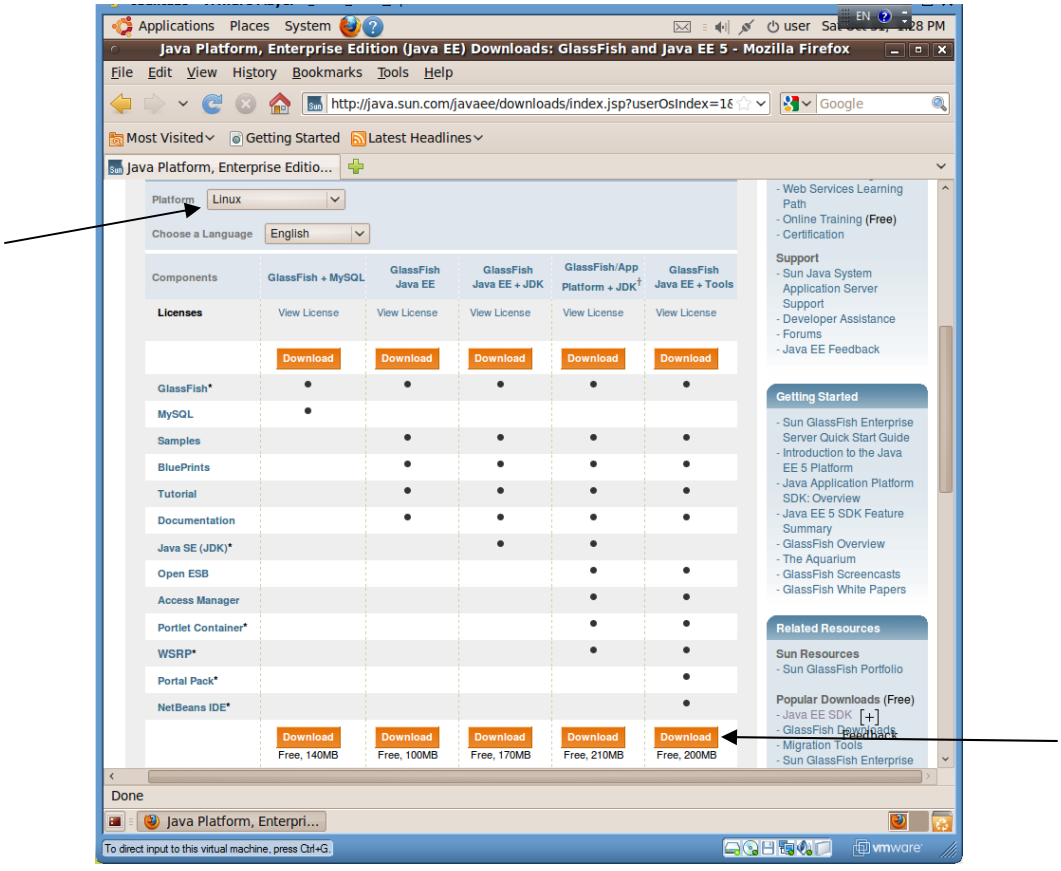
Instead of installing the basic *NetBeans* IDE, we will install Sun's Java EE (Enterprise Edition) bundle that includes *Netbeans* IDE, Sun's *GlassFish* application server, and example projects and documentation all integrated into a single package. Sun's *GlassFish* application server includes both a web server and an application server, and it will allow us to learn the advanced Java EE technologies based on *Enterprise JavaBeans* (EJBs).

### *4.12.1 Installing Java EE Bundle*

- Launch *Firefox* web browser and visit http://java.sun.com.

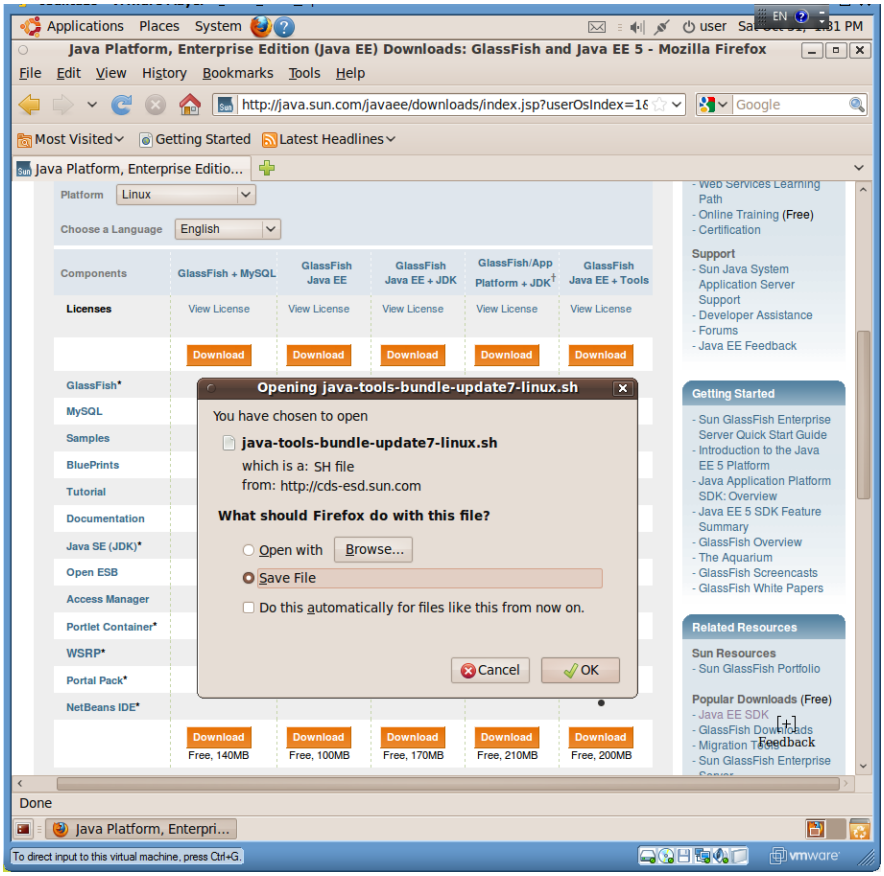



- Click on the “Java EE 5 SDK” link in the “Popular Downloads” pane.

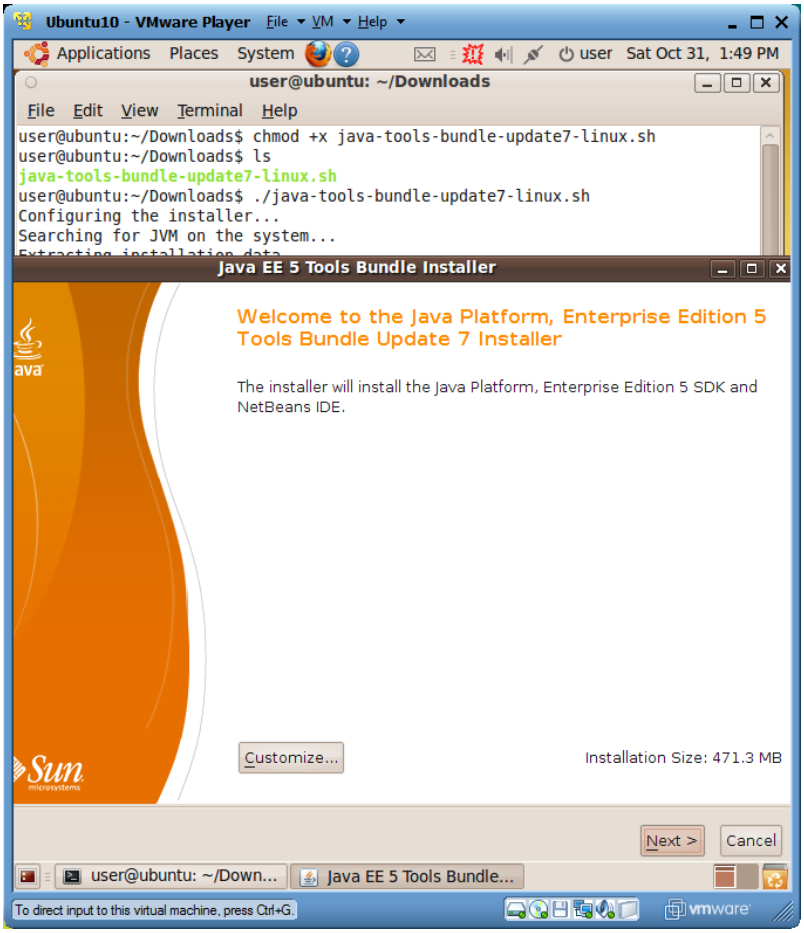



- Choose “Linux” for Platform. Click on the *Download* button at the bottom of the right-most column: GlassFish Java EE + Tools (200MB)

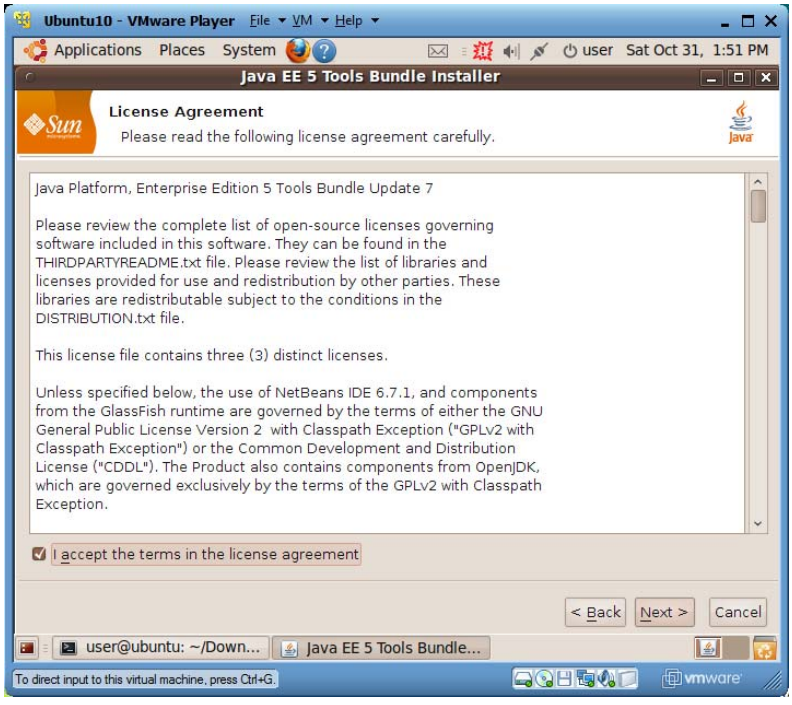
©Copyright 2009 Prof. Lixin Tao




- Check the “Save File” checkbox, and click on the *OK* button to start downloading the installer to “~/Downloads”.
- Start a terminal window, and run “cd ~/Downloads” to change working folder.
- Run command “chmod +x java-tools-bundle-update7-linux.sh” to make the installer executable.
- Use command “./java-tools-bundle-update7-linux.sh” to run the installer.

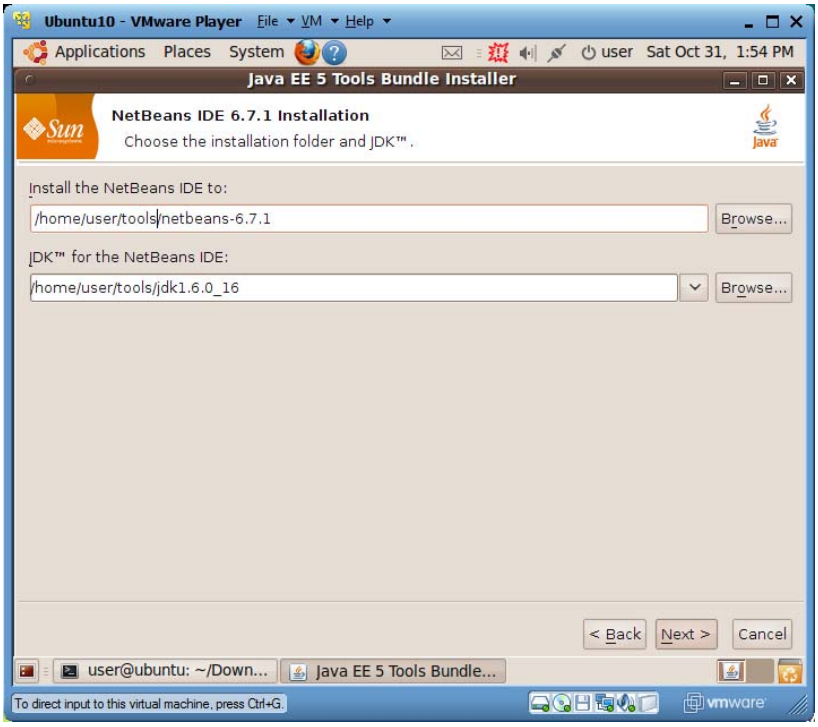



- Click on the *Next* button.

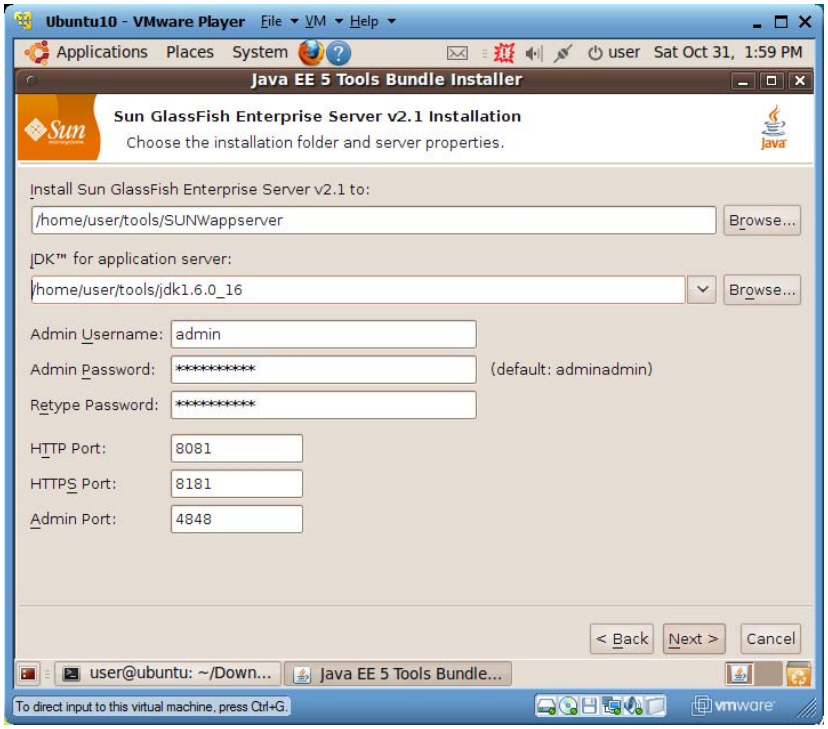


©Copyright 2009 Prof. Lixin Tao

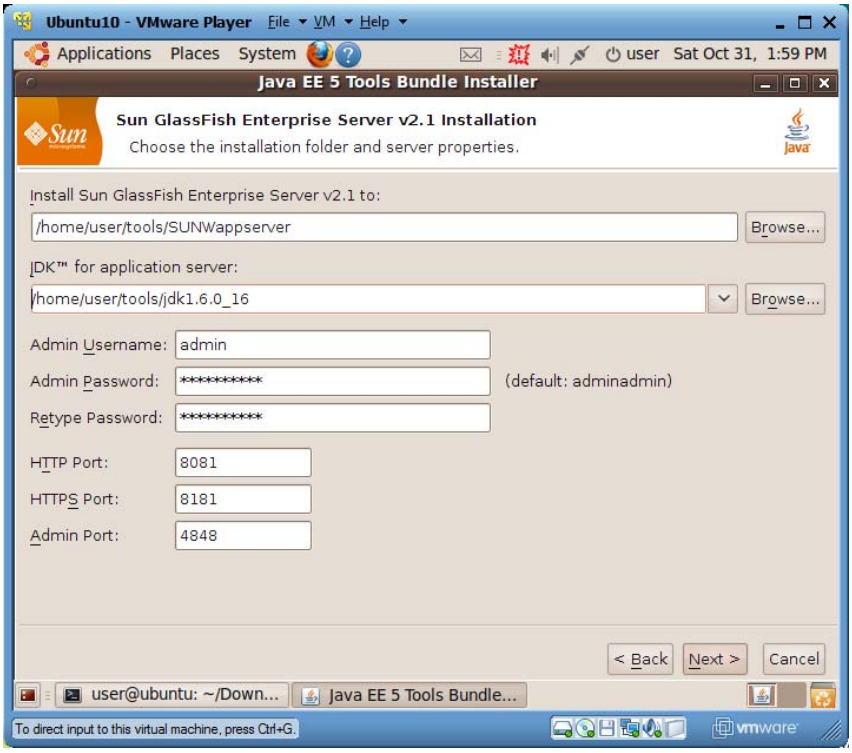
- Check “I accept ..” and click on the *Next* button.




- Use “/home/user/tools/netbeans-6.7.1” as the *NetBeans* IDE installation folder, and click on the *Next* button.



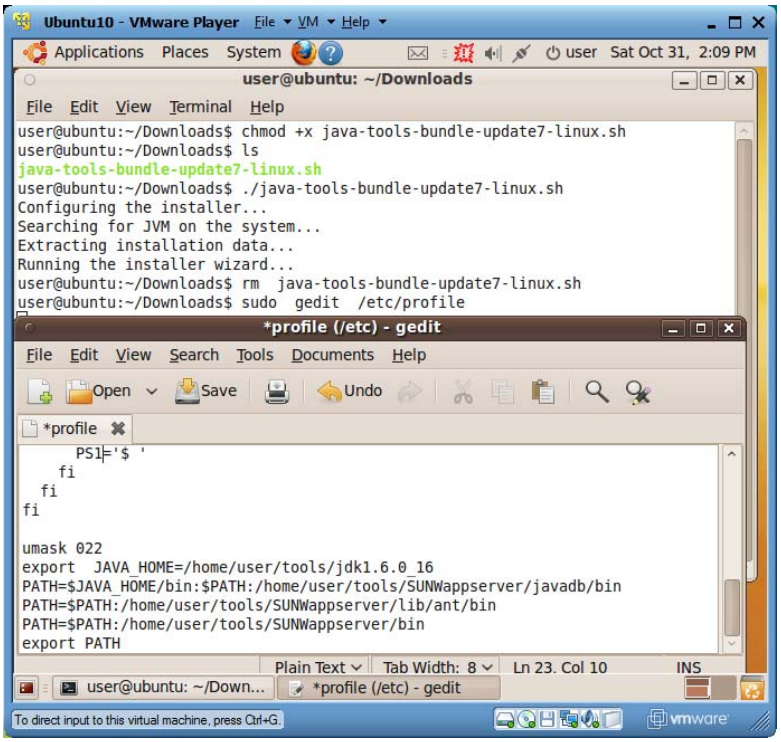
- Install Sun GlassFish server v2.1 in "/home/user/tools/SUNWappserver", accept and take a note of the default admin username "admin", password "adminadmin", HTTP Port 8081 (The default is 8080. Since we have *Tomcat* running, *GlassFish* now uses port 8081. You need to change the port setting for some example web applications.), HTTPS port 8181, and Admin Port 4848. Then click on the *Next* button.




- Click on the *Install* button to complete the installation. The installer will add a "NetBeans IDE 6.7.1" shortcut on the Desktop.
- After the installation wizard comes to the end, you may delete the installer by running command "rm java-tools-bundle-update7-linux.sh" in "~/Downloads".
- Run "sudo gedit /etc/profile" to update the PATH definition as below (all on the same line; no line breaks) to add "/home/user/tools/SUNWappserver/javadb/bin", "/home/user/tools/SUNWappserver/lib/ant/bin" and "/home/user/tools/SUNWappserver/javadb/bin" to PATH.

```
PATH=$JAVA_HOME/bin:$PATH:/home/user/tools/SUNWappserver/javadb/bin
PATH=$PATH:/home/user/tools/SUNWappserver/lib/ant/bin
PATH=$PATH:/home/user/tools/SUNWappserver/bin
export PATH
```

©Copyright 2009 Prof. Lixin Tao



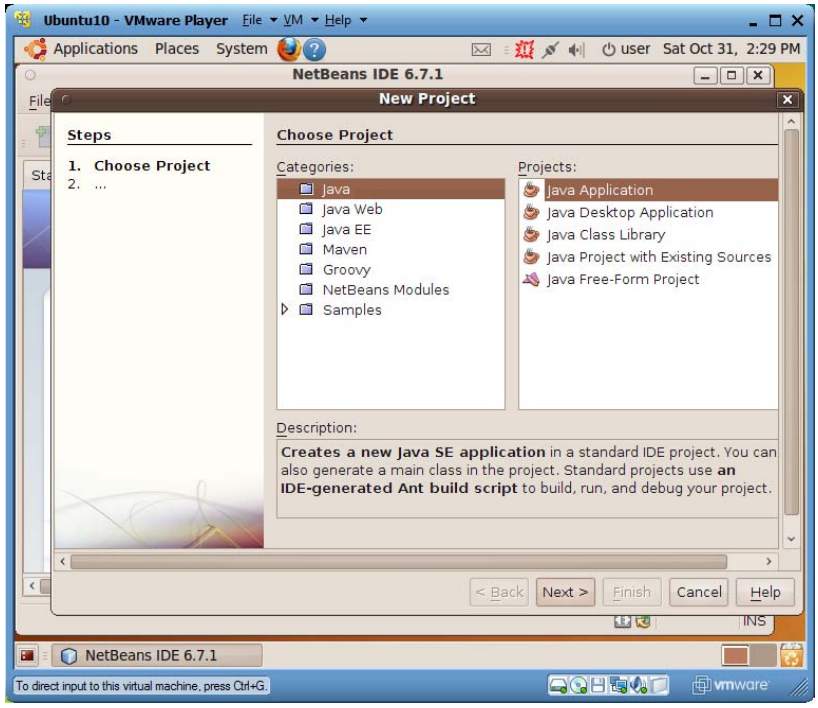

- Run command "gedit ~/.bashrc" to define an alias (nickname) for *NetBeans* and define a new variable DERBY_HOME (Sun database), as shown below:

```
export DERBY_HOME=~/tools/SUNWappserver/javadb
alias  netbeans='~/tools/netbeans-6.7.1/bin/netbeans&'
```

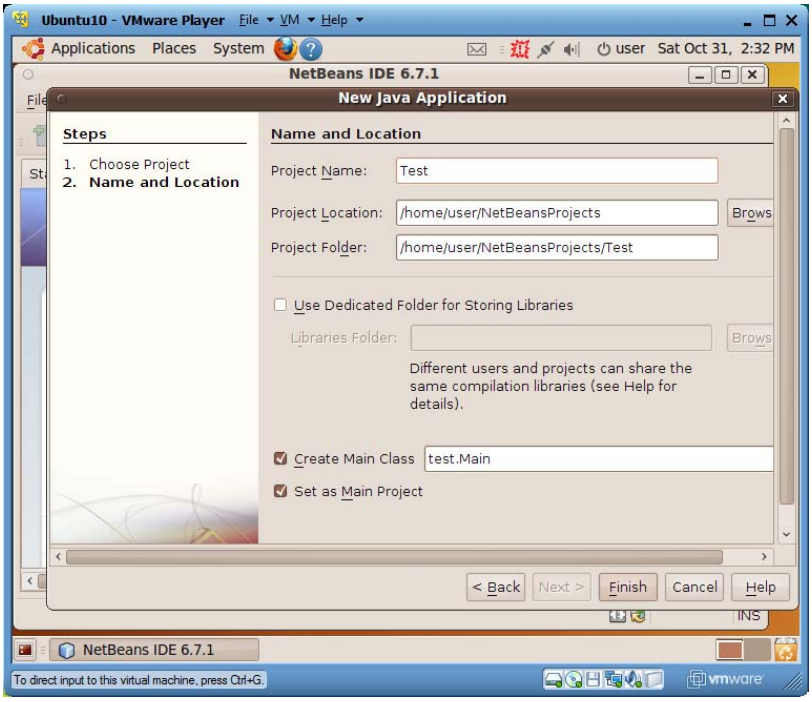
- Reboot the Linux PC.
- Now you can run command "netbeans" to launch *NetBeans* IDE. You can also use menu item "Applications|Programming|NetBeans IDE 6.7.1" to launch *NetBeans* IDE (you will see this option after you have restarted your VM). As the third option you can also double-click on the "NetBeans IDE 6.7.1" shortcut on *Desktop* to launch *NetBeans* IDE.

### *4.12.2 Developing a Test Java Program*

- If you have not launched *NetBeans* IDE yet, start it by executing command "netbeans" in a terminal window.
- Click on menu item "File|New Project …" to launch the "New Project" window. Make sure *Categories* selection chooses "Java", and Projects selection chooses "Java Application", as shown below.

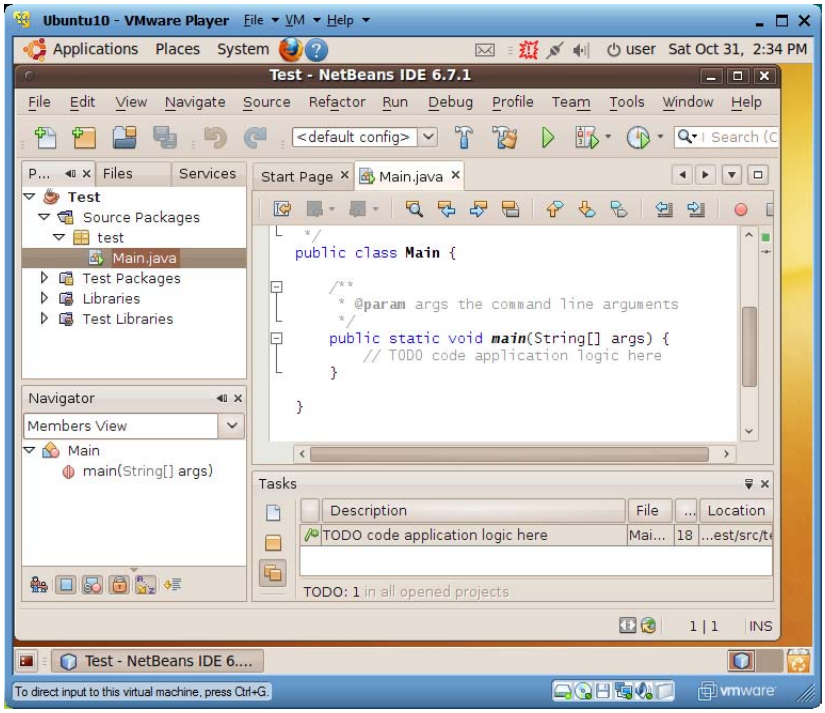



- Click on the *Next* button and see the "New Java Application" window. Type "Test" in the "Project Name" text field, as shown below. Make sure that the checkboxes for "Create Main Class" and "Set as Main Project" are checked.

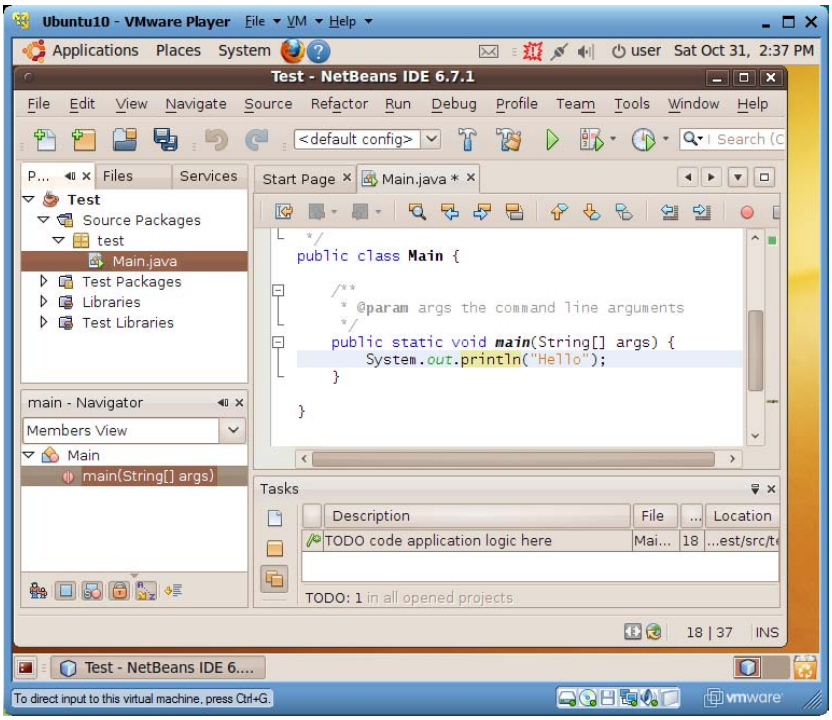



- Click on the *Finish* button, and you will see a screen similar to the following one:

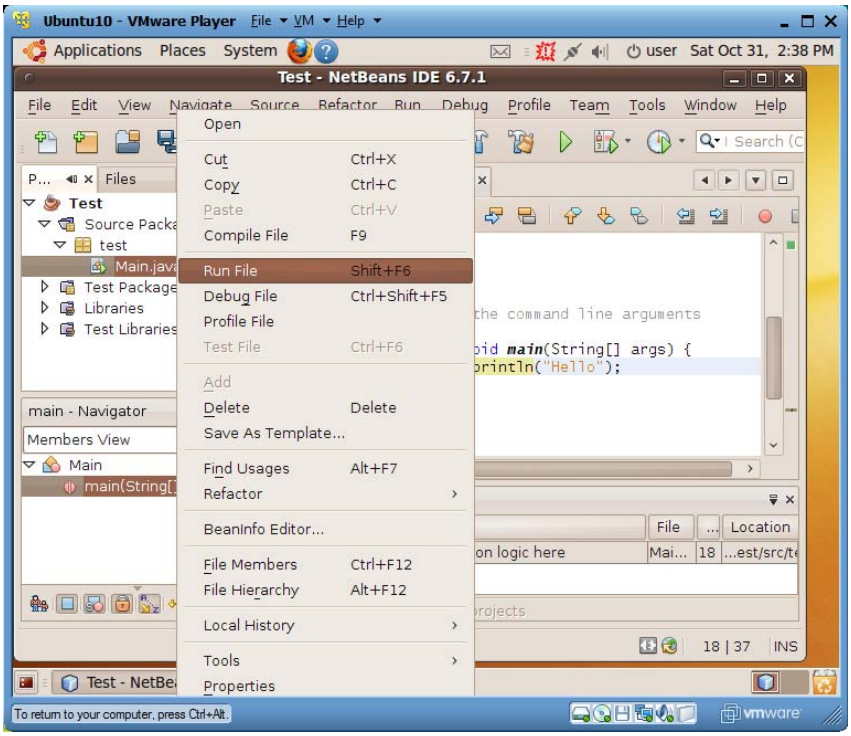
©Copyright 2009 Prof. Lixin Tao



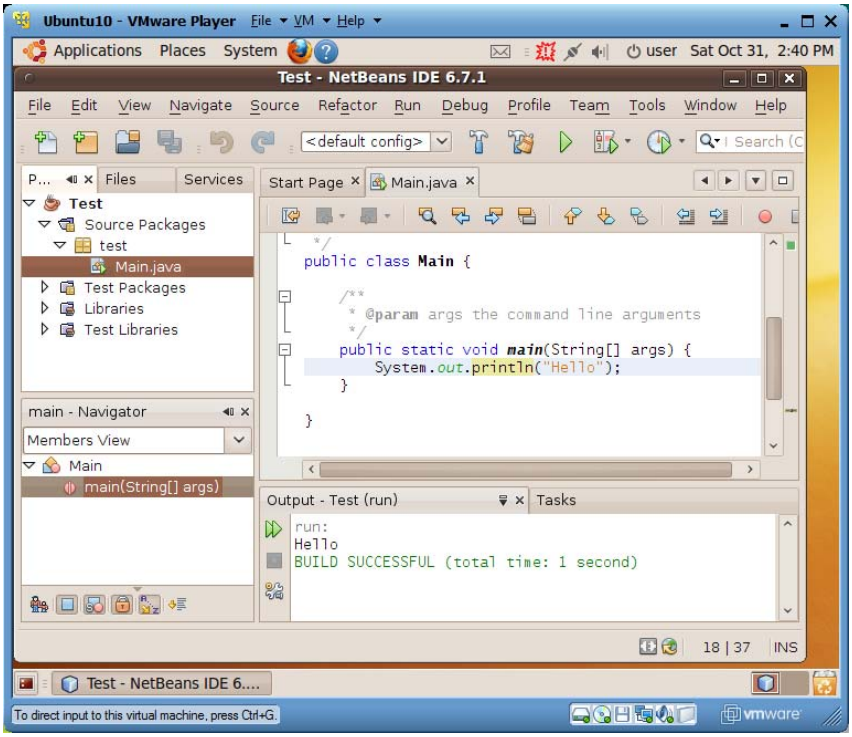

- In the body of method main, enter “System.out.println(“Hello”);”, as shown below.




- In the left-upper project pane, right-click on file “Main.java” and choose “Run File” on the popup menu, as show below:




- The program will be saved to hard disk, compiled into a bytecode file, and executed. The execution output is displayed under the “Output” tab, as shown below:




- Click on menu item “File|Close Project” to close this project, and use “File|Exit” to shut down *NetBeans* IDE.

©Copyright 2009 Prof. Lixin Tao

- Your *NetBeans* projects are saved under "/home/user/NetBeansProjects".

### *4.12.3 Starting/Stopping GlasshFish Application Server and Derby Database Server*

- To start *GlassFish* application server, run "asadmin start-domain --verbose domain1". The terminal window should seem hang and not usable when the server is running. It is normal. The server error/debug messages and printout from the server applications will be printed in this terminal window. Don't shut down this window; instead minimize it. Otherwise the server would shutdown. You can use the web browser to visit http://localhost:8081 to check whether *GlassFish* is running.
- To stop *GlassFish* application server, run "asadmin stop-domain domain1" from another terminal window.
- To start *Derby* database server, run "asadmin start-database"
- To stop *Derby* database server, run "asadmin stop-database".

If you use *NetBeans* IDE to develop web applications, these two servers will start and stop automatically as needed.

For convenience, let us define some aliases for starting and stopping *GlassFish* and *Derby* servers.

- Run "gedit ~/.bashrc" to add the following lines at the end of file "~/.bashrc":

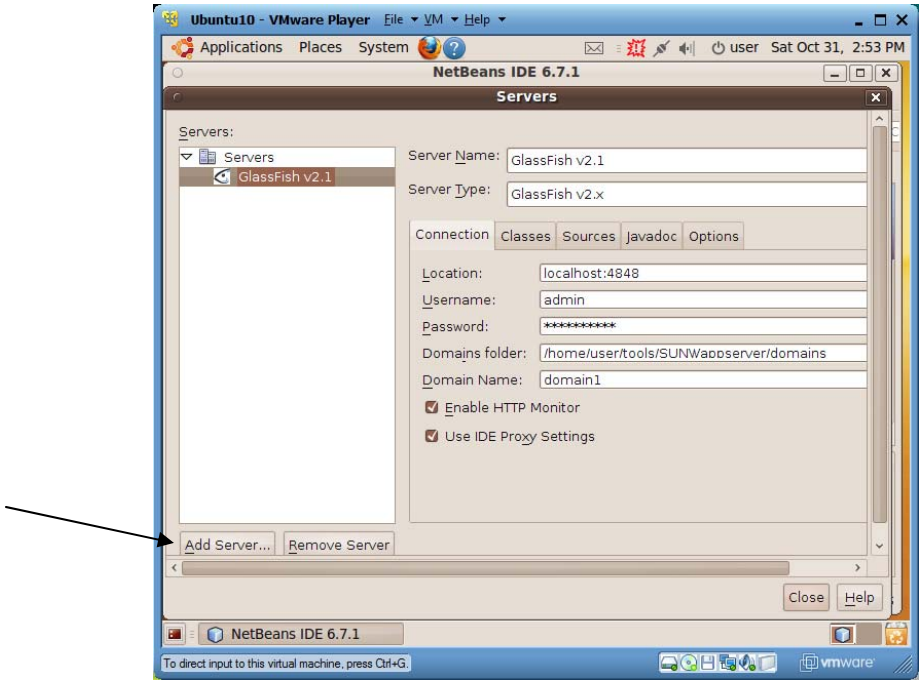
```
alias glassfish-start='asadmin start-domain --verbose domain1'
alias glassfish-stop='asadmin stop-domain domain1'
alias derby-start='asadmin start-database'
alias derby-stop='asadmin stop-database'
```

- Run "source ~/.bashrc" or reboot your Linux PC.
- From now on you can run "glassfish-start" to launch *GlassFish* application server, run "glassfish-stop" to stop it; run "derby-start" to launch *Derby* database server, and run "derby-stop" to stop it.

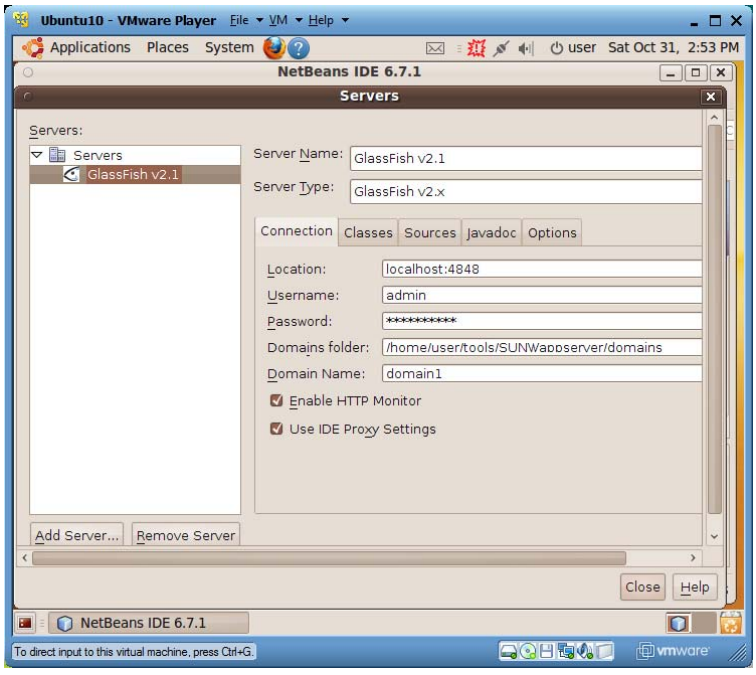
### *4.12.4 Adding Tomcat Server to NetBeans*

To use *NetBeans* to develop web applications on our *Tomcat* web server, we need to add the *Tomcat* to *NetBeans*.

- Launch *NetBeans* by running "netbeans".
- Click on menu item "Tools|Servers" to launch the "Servers" window.
- Check the "Enable HTTP Monitor" checkbox for *GlassFish*.

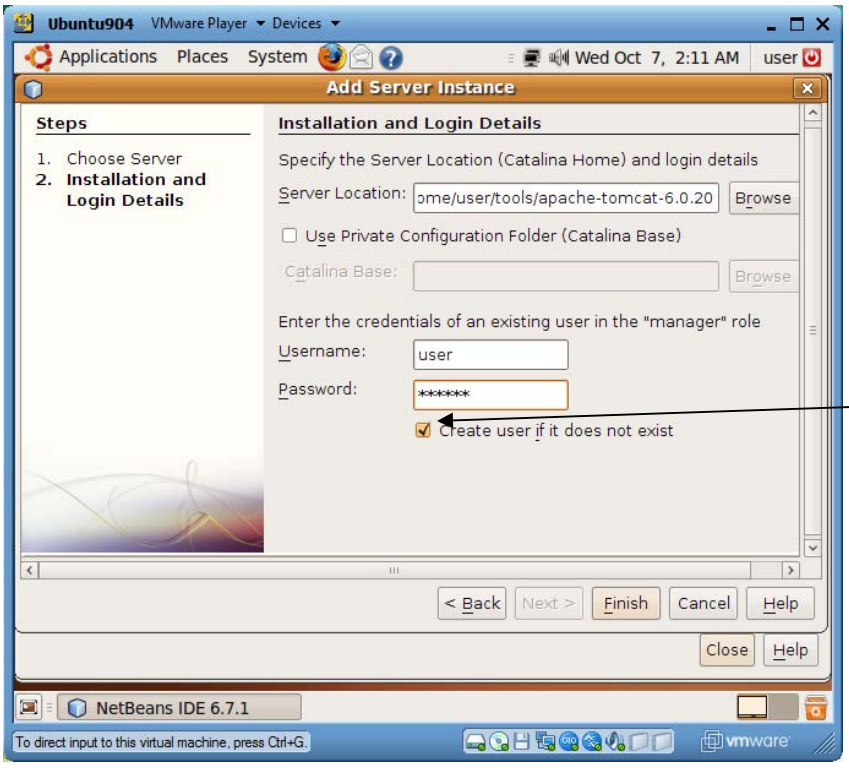



- Click on the “Add Server” button to launch the “Add Server Instance” window. Choose “Tomcat 6.0” in the “Choose Server” pane.

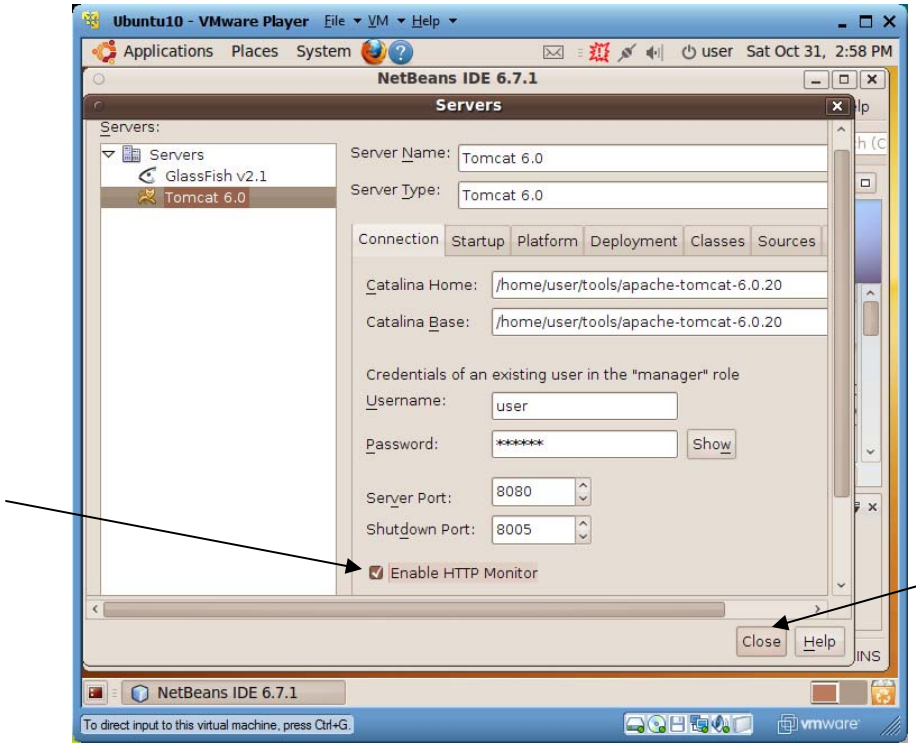



- Click on the *Next* button.

©Copyright 2009 Prof. Lixin Tao




- Use *Browse* to enter "/home/user/tools/apache-tomcat-6.0.20" for *Catalina Home* (*Tomcat* installation base folder), "user" for *Username*, and "123456" for *Password*, check the "Create user if it does not exist" checkbox, and click on the *Finish* button.



- Check the checkbox for “Enable HTTP Monitor” in the “Servers” window, and click on the *Close* button to complete the adding server operation.
- Now you are ready to use *NetBeans* IDE to develop web applications on our *Tomcat* server as well as on *GlassFish* application server.

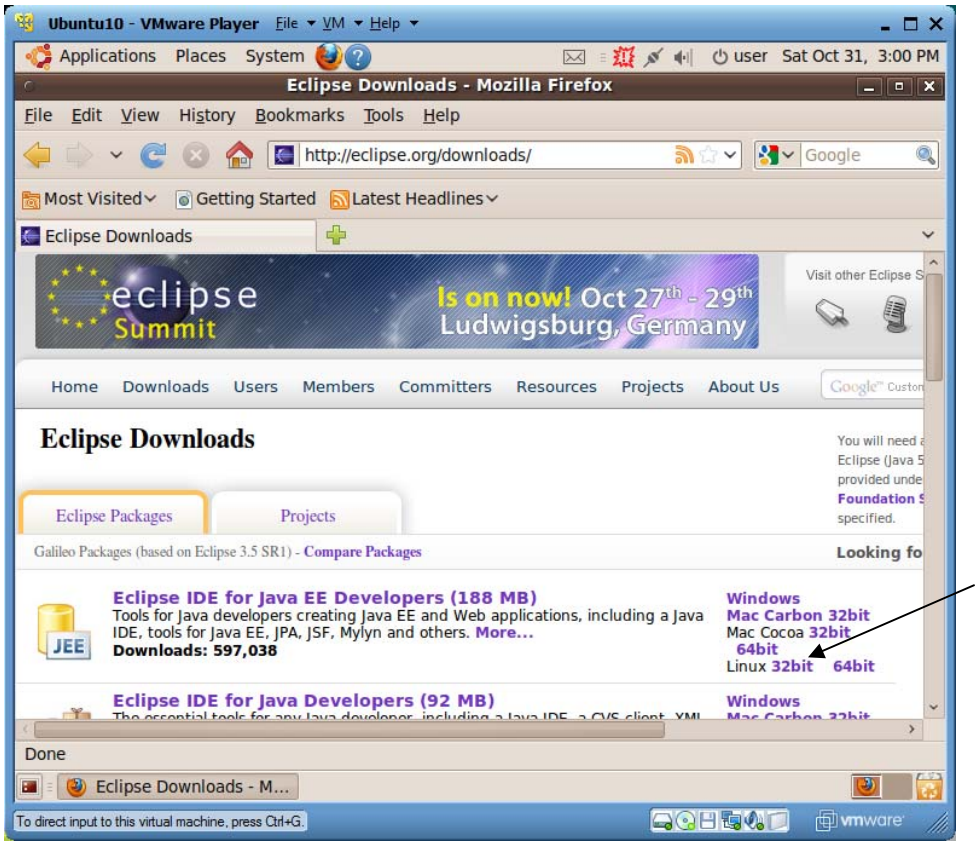
## 4.13 Installing *Eclipse*

Before we can install *Eclipse*, we must have installed Java JDK.

*Eclipse* is another IDE (Integrated Development Environment) for developing software projects, mainly for Java and Java-based web applications. Since *Eclipse* is implemented in Java itself, Java JRE or JDK must be installed before this step.

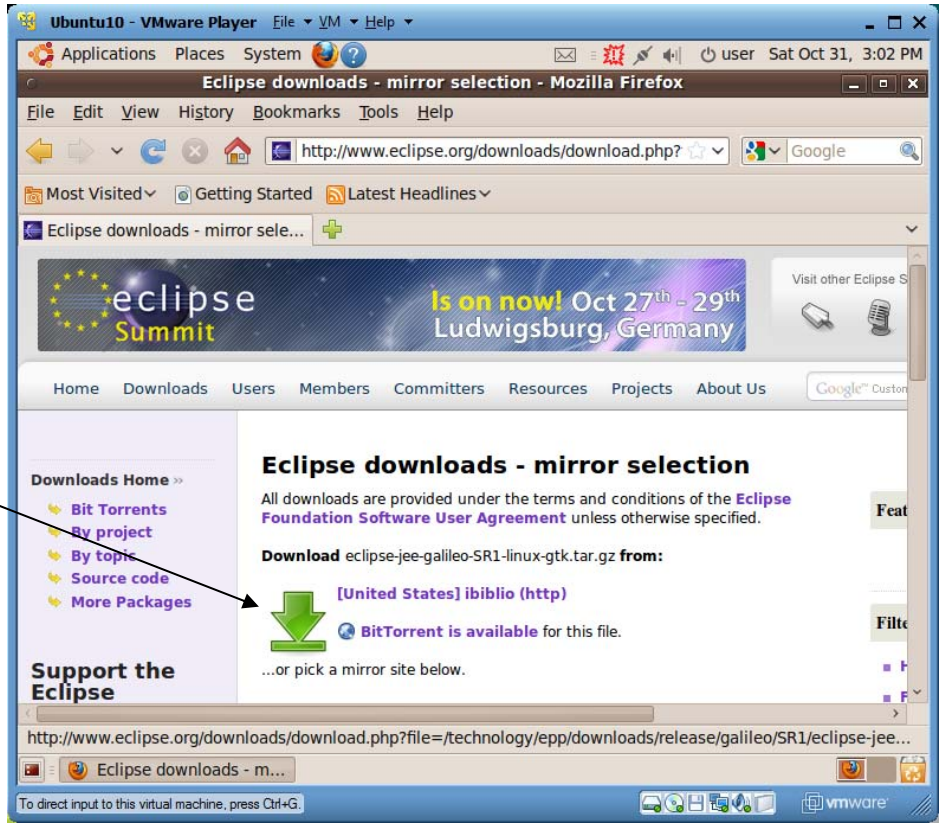
### *4.13.1 Downloading Eclipse Installer*

- Launch *Firefox* web browser and visit http://eclipse.org/downloads. Under the *Eclipse Packages* tab, in the section for “Eclipse IDE for Java EE Developers (189MB)”, click on the link of *32bit* after “Linux” to download the installer.

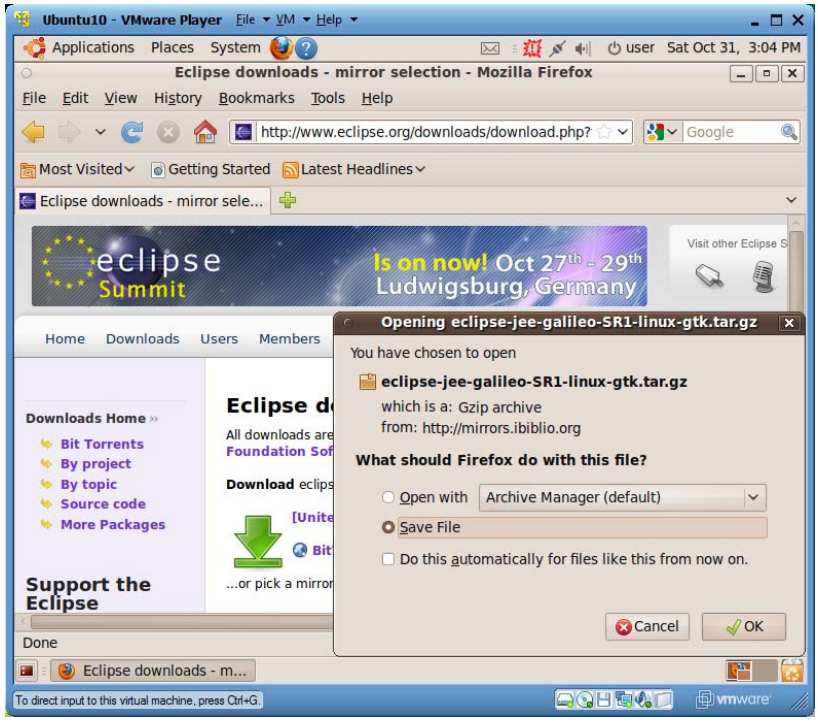



- Click on the big green download arrow on the next screen capture to start downloading the installer to your folder *Downloads*.

©Copyright 2009 Prof. Lixin Tao



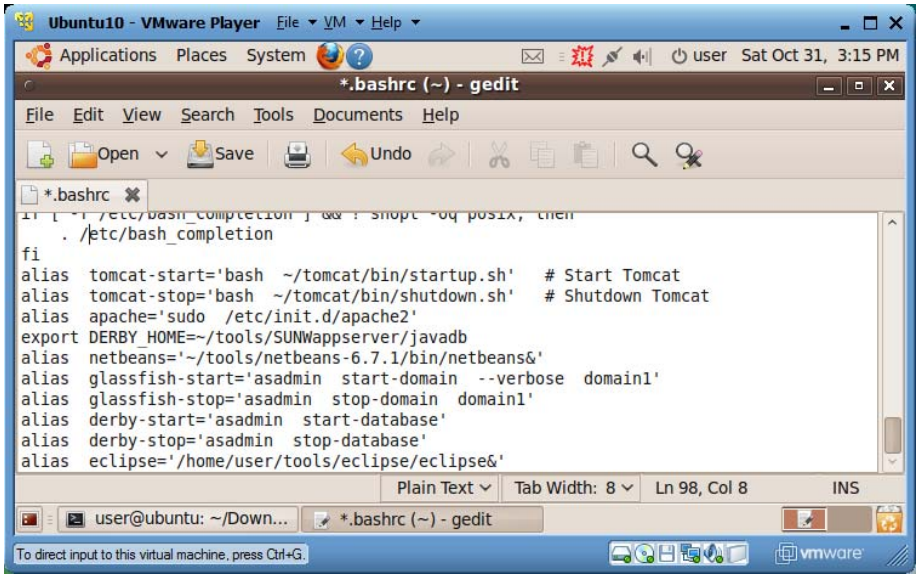

- Check the "Save File" checkbox, and click on the *OK* button to save the installer to your folder *Downloads*.



### *4.13.2 Installing Eclipse*

- Launch a terminal window, and run "cd ~/Downloads" to open a terminal window in "~/Downloads".
- Run command "tar xvzf eclipse-jee-galileo-SR1-linux-gtk.tar.gz -C ~/tools" to unzip the installer contents in "~/tools". The *Eclipse* installation folder is "/home/user/tools/eclipse".
- You could delete the installer by running command "rm eclipse-jee-galileo-SR1-linux-gtk.tar.gz" in "~/Downloads".
- Run command "gedit ~/.bashrc" to insert the following line at the end of file "~/.bashrc" to define an alias "eclipse" for launching *Eclipse*.

```
alias eclipse='/home/user/tools/eclipse/eclipse&'
```

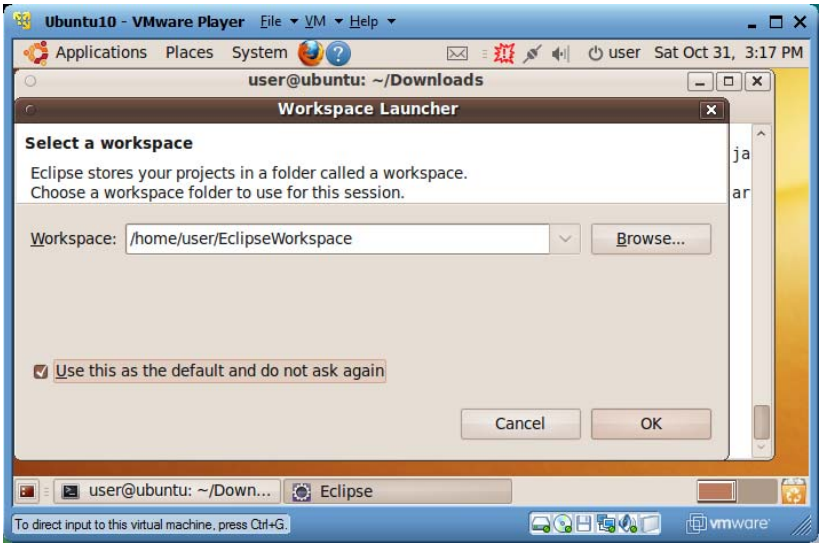



- Run command "source ~/.bashrc" to process the definitions in "~/.bashrc".
- To run *Eclipse*, just run command "eclipse".

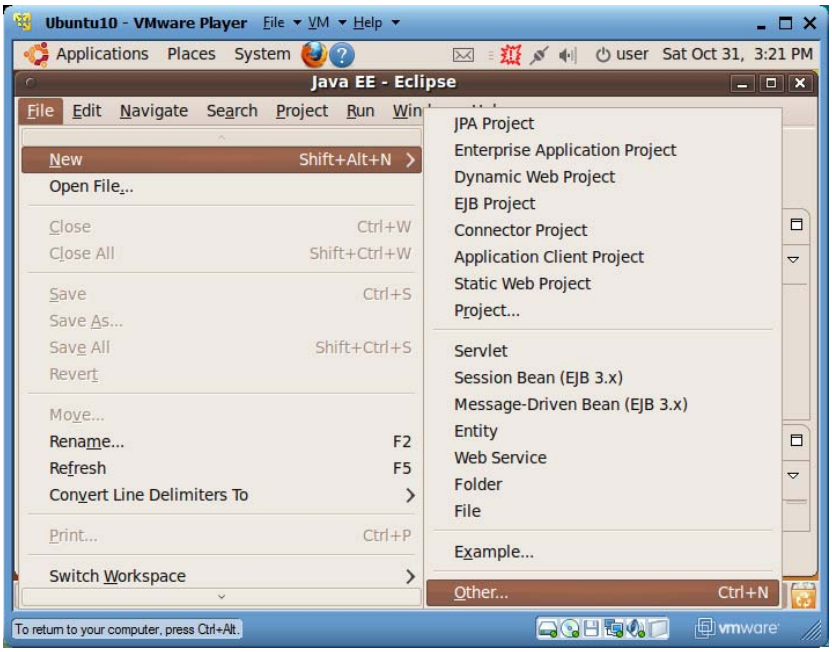
### *4.13.3 Developing a Sample Java Program*

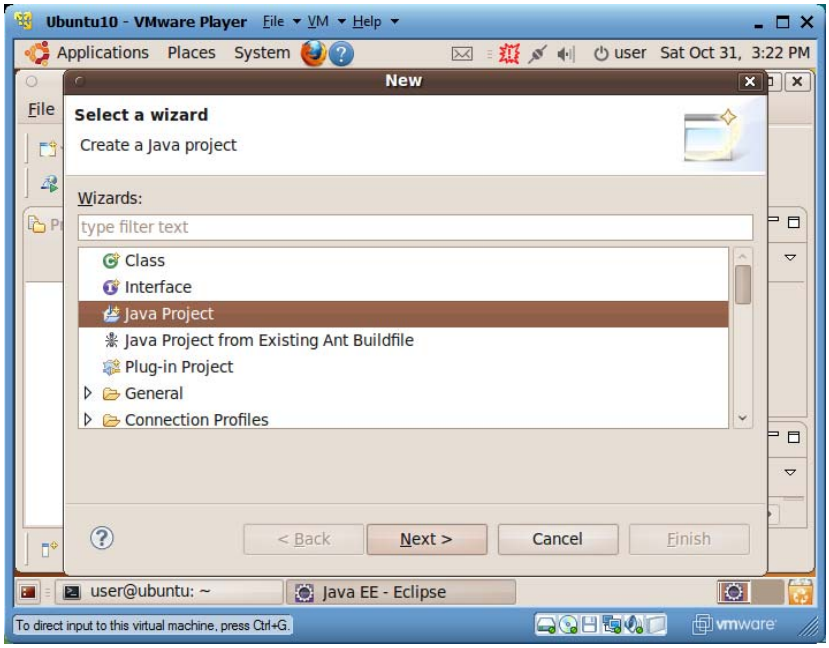
- When you run *Eclipse* the first time, use "/home/user/EclipseWorkspace" as the default project workspace, and check the checkbox for "Use this as the default and do not ask again".

©Copyright 2009 Prof. Lixin Tao

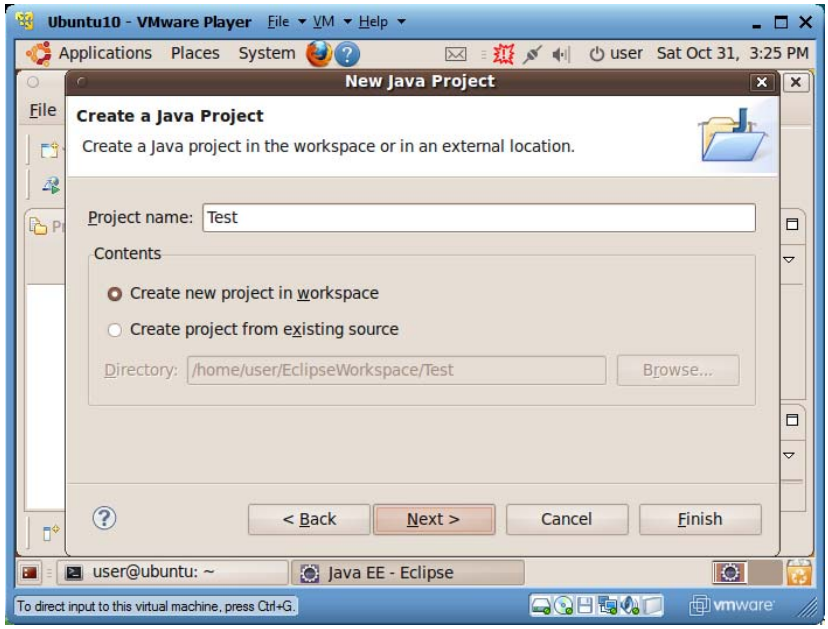



- When creating a Java project, first choose "File|New|Other", then choose "Java Project" to switch to Java Perspective and add "Java Project" to the *New Project* menu.



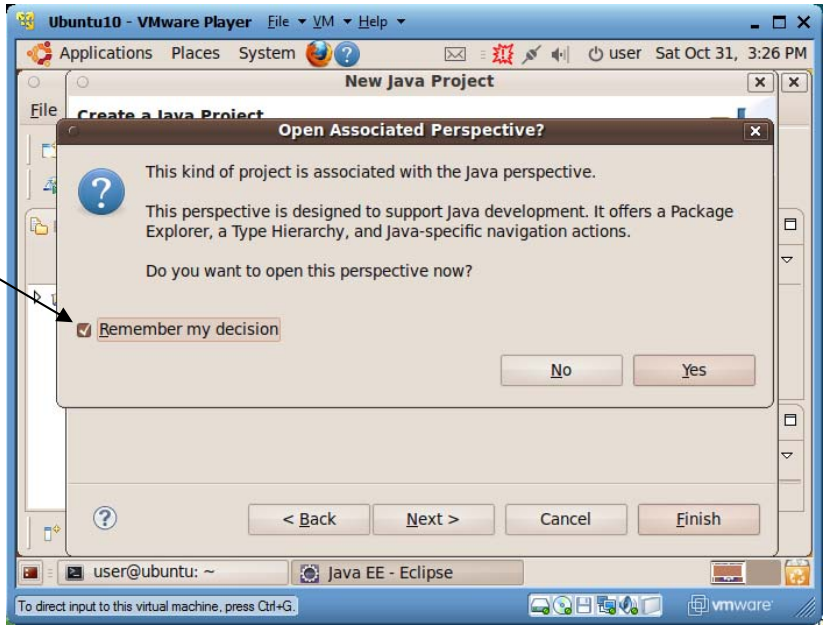



- Click on the *Next* button to get the following view, and enter “Test” in “Project Name” text field. Let the “JRE” and “Project layout” panes keep their default values.

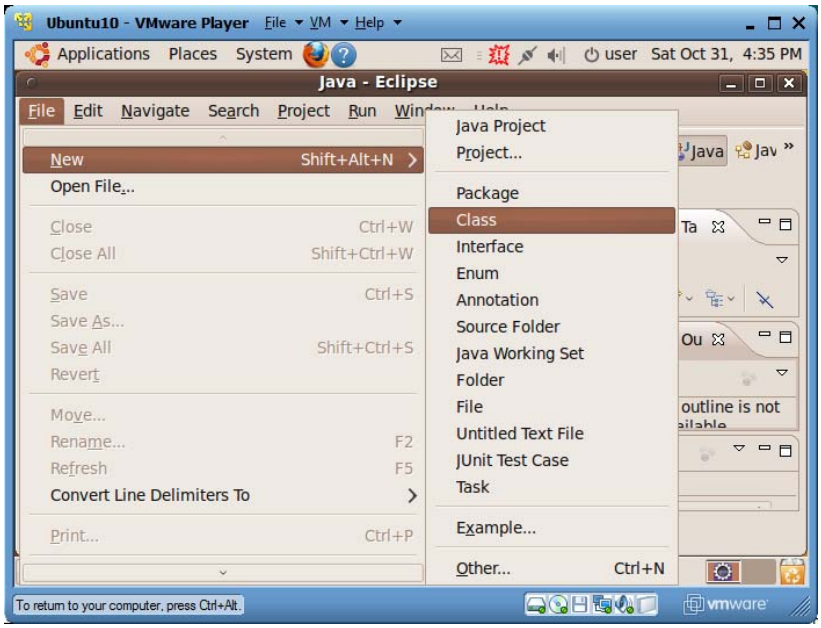



- Click the *Finish* button to create the “Test” Java project. You will then see the following screen:

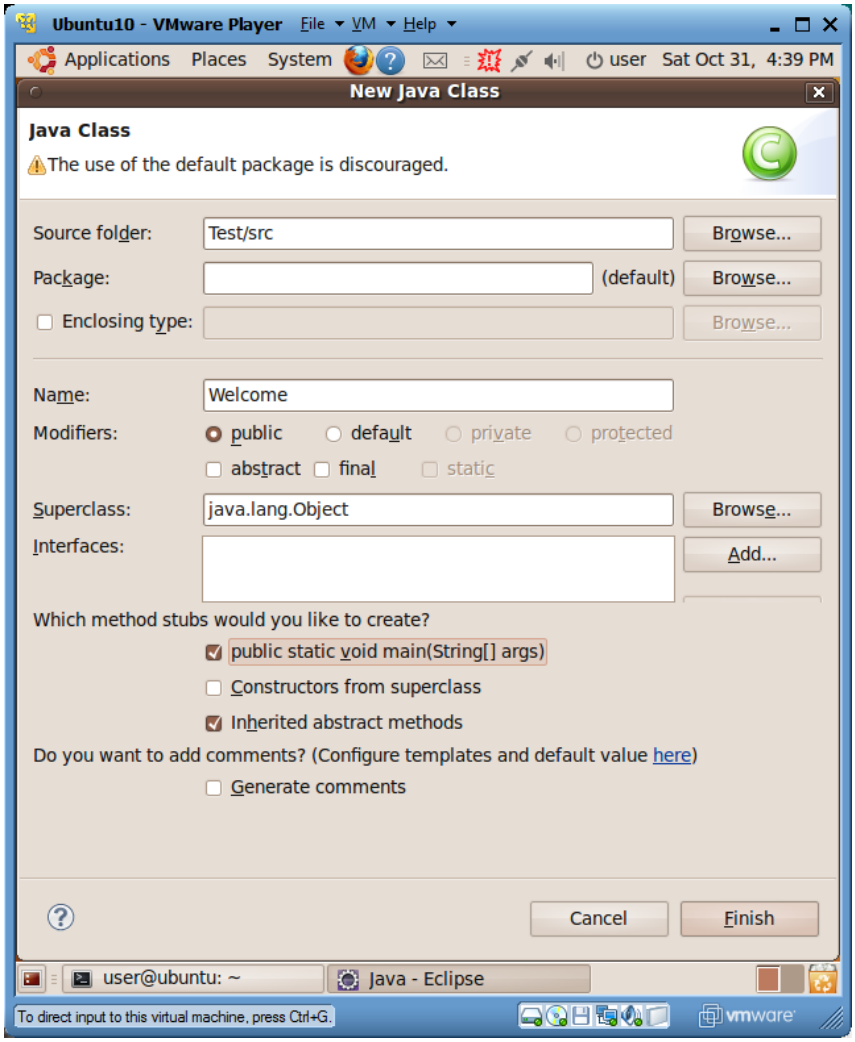
©Copyright 2009 Prof. Lixin Tao




- Check the "Remember my decision" checkbox, and click on the *Yes* button. From now on you have "Java Project" on your "File|New" menu.
- Click on "File|New|Class" menu item to create a new Java class source file.

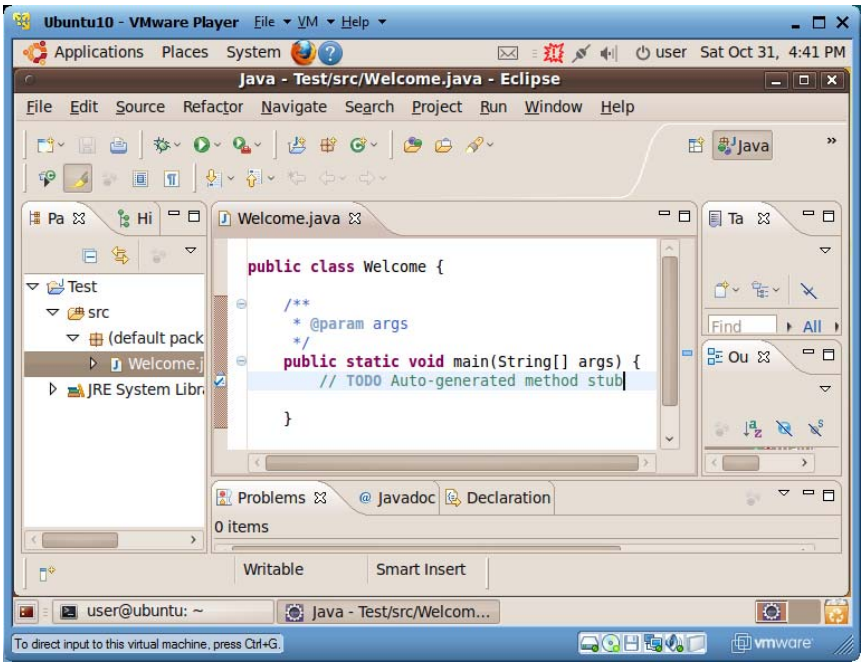



- In the following "New Java Class" window, enter "Welcome" as the class name. Check the "public static void main" checkbox to create the "main" method in the class.

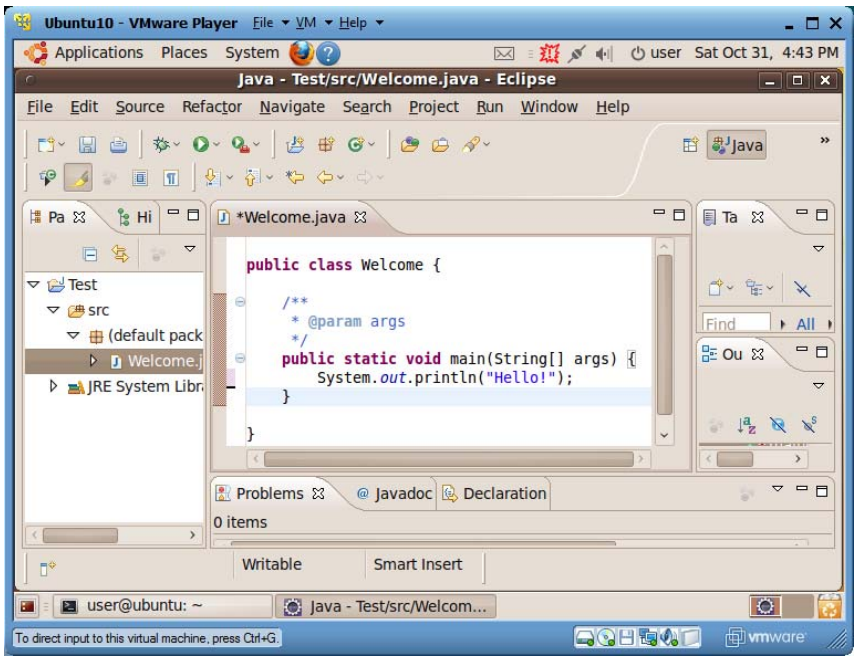



- Click on the *Finish* button and you will have the following view:

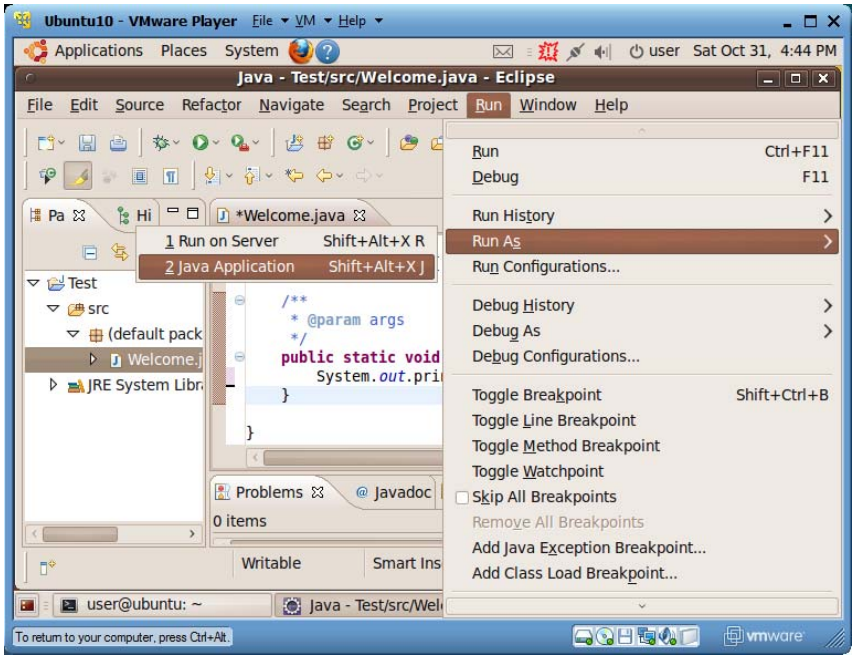
©Copyright 2009 Prof. Lixin Tao



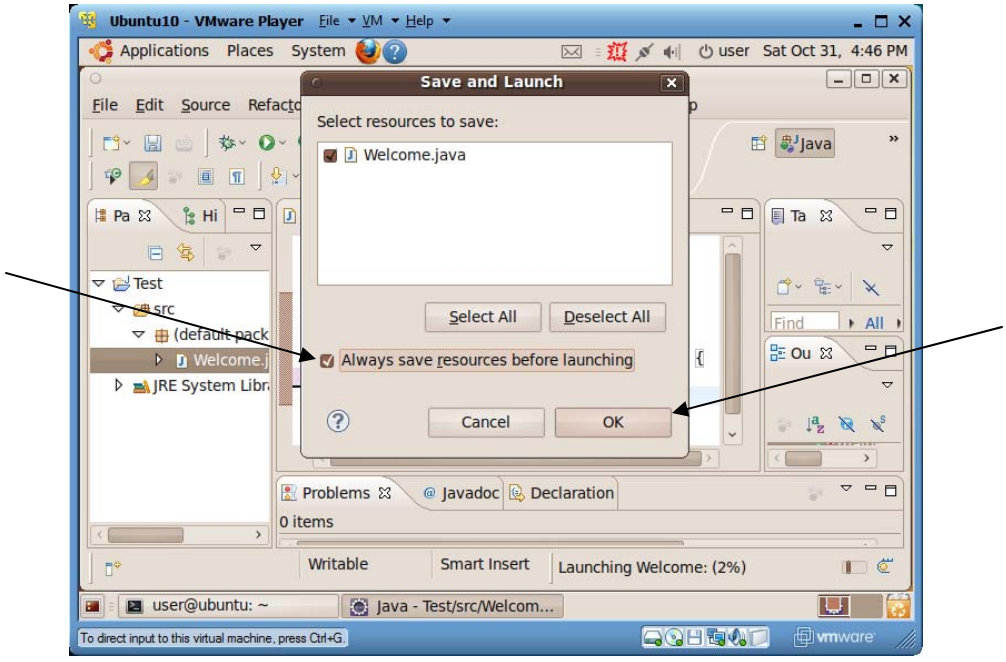

- In the body of method "main", replace the TODO comment with Java statement "System.out.println("Hello!");" as shown below.




- Click on menu item "Run|Run As|Java Application" to compile and run the project.

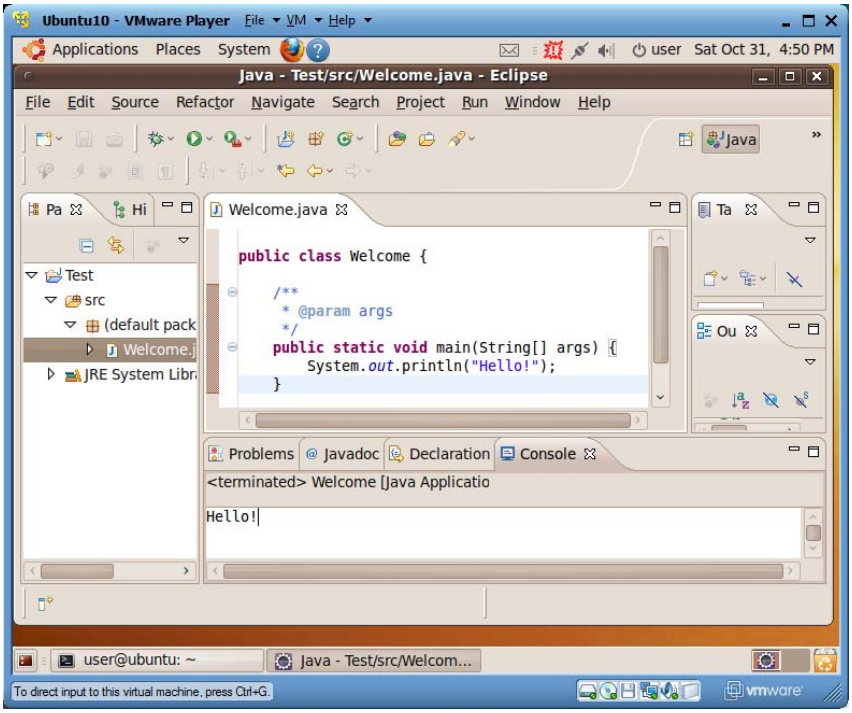



- *Eclipse* now asks you to confirm to save and launch the project. Check both of the two checkboxes as shown below:




- Click on the *OK* button, and *Eclipse* save the modified source file, compile it and run it. The execution output is under the “Console” tab, as shown below:

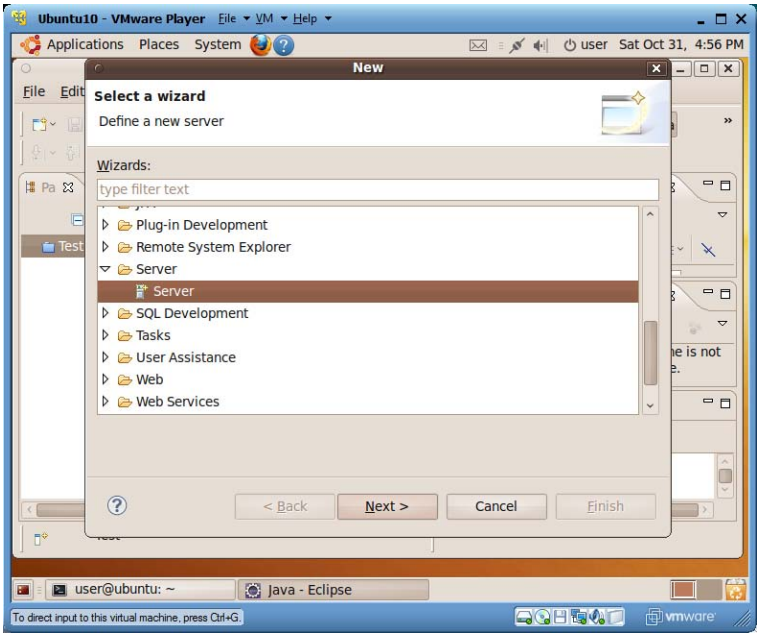
©Copyright 2009 Prof. Lixin Tao



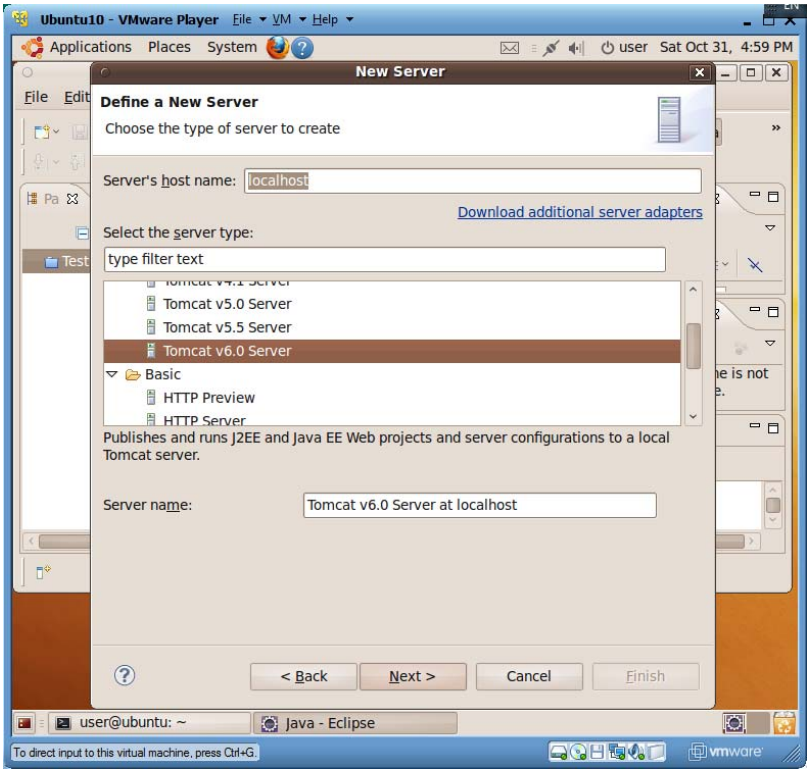

- You can use “File|Close” to close the current project, and “File|Exit” to shut down the *Eclipse* IDE.

### *4.13.4 Adding Tomcat Server to Eclipse*

- Run “eclipse” to launch *Eclipse* IDE
- Choose “File|New|Other…”, and then choose *Server* in the *New* window, as shown below:

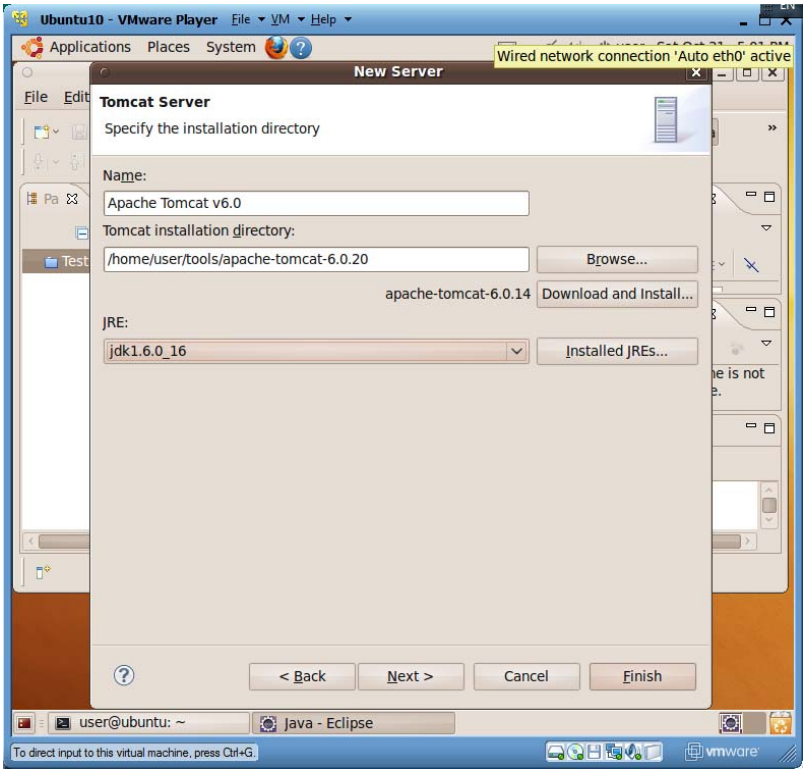


- Click on the *Next* button, and choose “Tomcat v6.0 Server” as shown below.

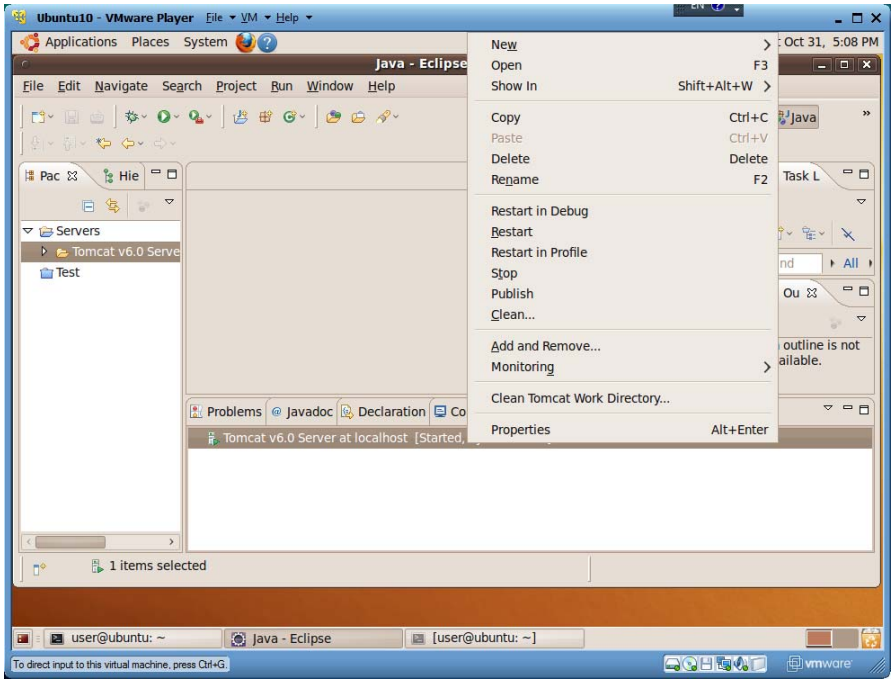



- Click on the *Next* button, enter “/home/user/tools/apache-tomcat-6.0.20” (you can browse for it) in the “Tomcat installation directory” text filed, and choose “jdk1.6_16” for JRE.

©Copyright 2009 Prof. Lixin Tao



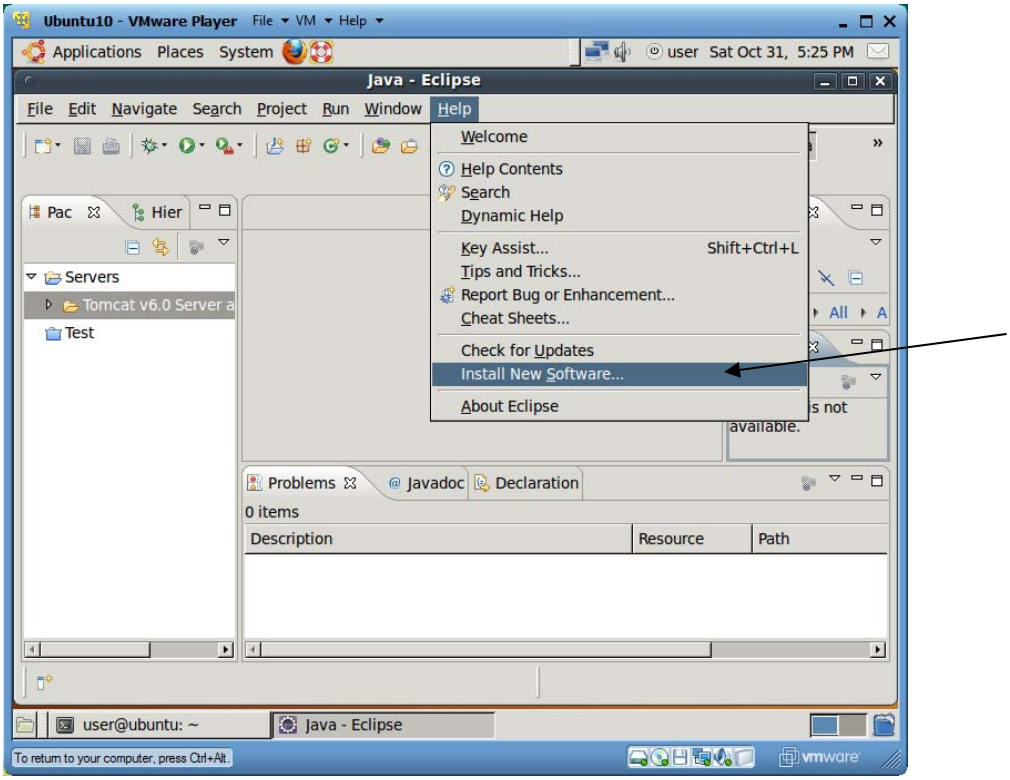

- Click on the *Finish* button to complete the addition of our *Tomcat* in *Eclipse*. You can right-click on the Tomcat server item under the "Servers" tab and use the popup menu to control the start and stop of the *Tomcat* web server for testing web applications inside *Eclipse* (you may not be able to use the *Tomcat* outside of *Eclipse*). If *Tomcat* is running outside of *Eclipse*, you need to run "tomcat-stop" to stop it before you can start *Tomcat* in *Eclipse*.



### *4.13.5 Adding GlassFish Server to Eclipse*

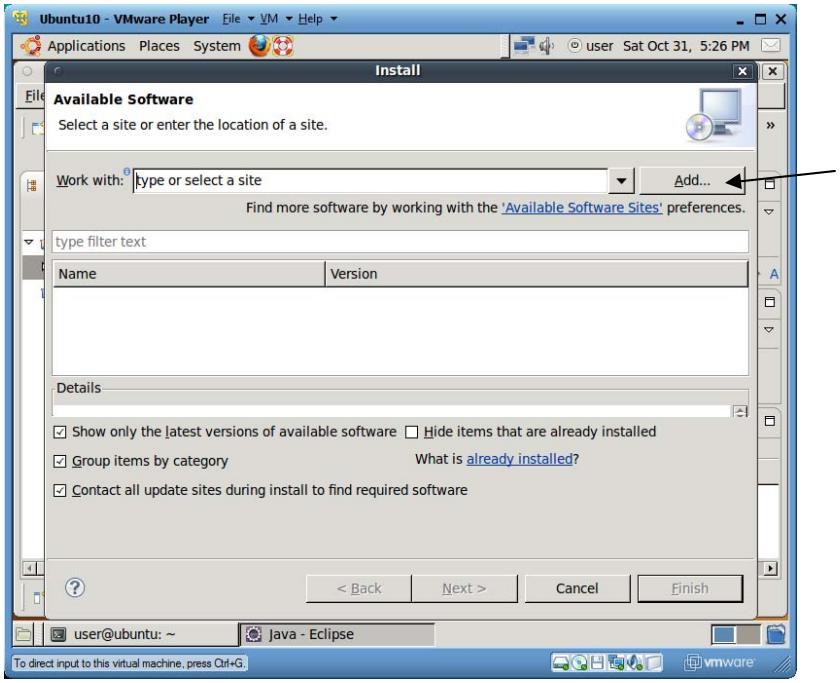
*GlassFish* application server is installed when we install the Java EE bundle including *NetBeans* as described in the last section. Make sure you have installed the Java EE bundle by now so *GlassFish* is already installed.

- Run “eclipse” to launch *Eclipse* IDE
- Click on menu item “Help|Install New Software…”

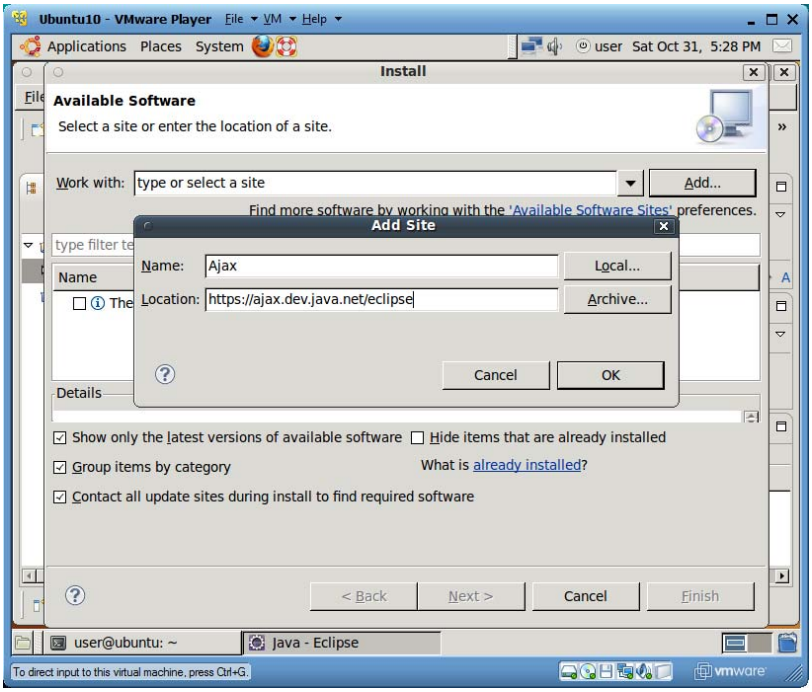



- The following *Install* window will pop up.

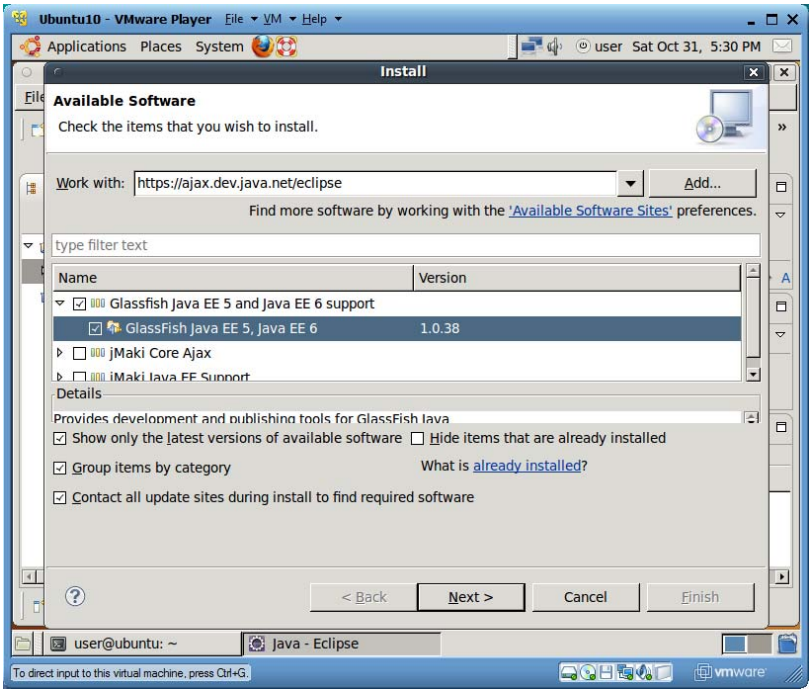
©Copyright 2009 Prof. Lixin Tao



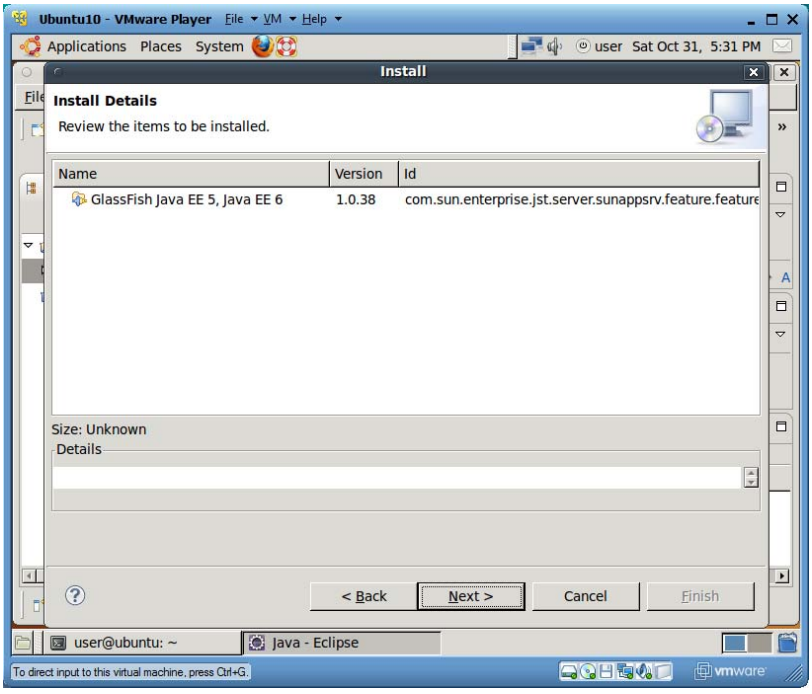

- Click on the "Add…" button, then in the pop up "Add Site" window, enter "Ajax" for *Name*, and https://ajax.dev.java.net/eclipse for *Location*, as shown below.




- Click on the *OK* button. Wait for a few minutes until "Glassfish Java EE" is listed under "Name", as shown below.

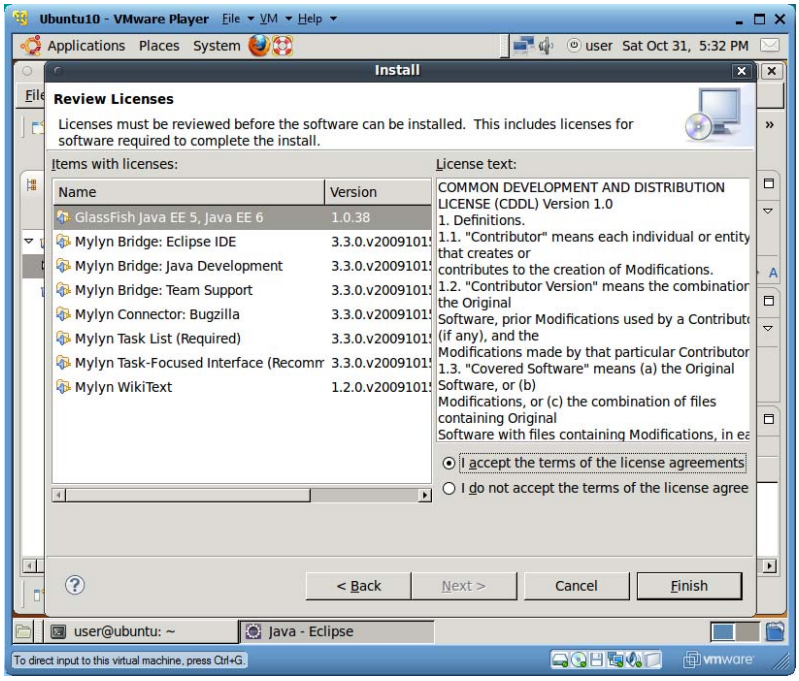



- Check for “Glassfish Java EE 5 and Java EE 6” and click on the *Next* button.

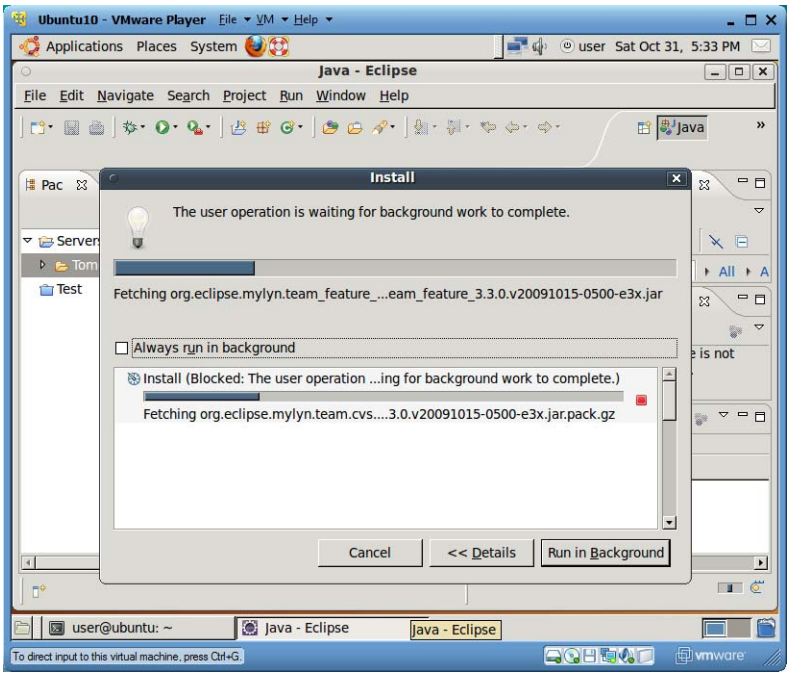



- Click on the *Next* button.

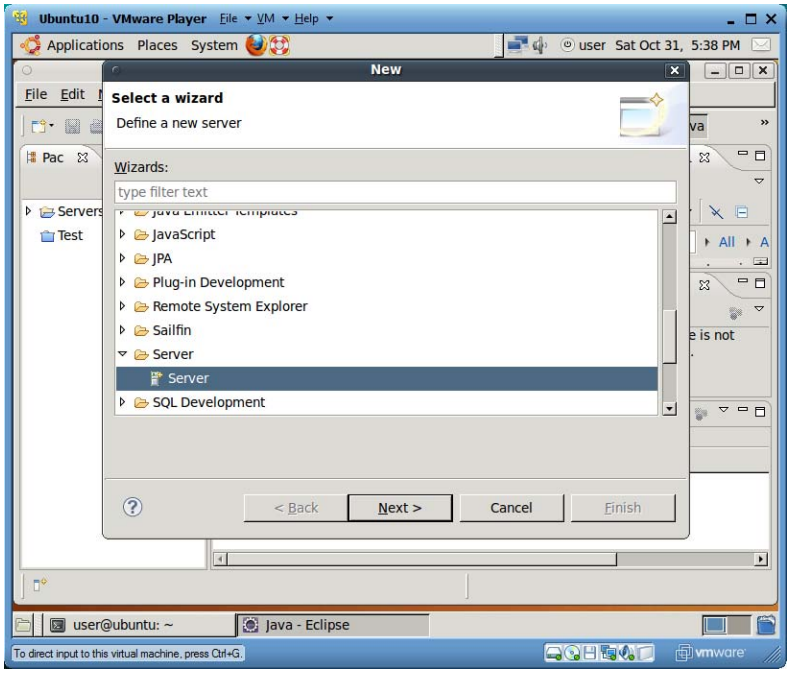
©Copyright 2009 Prof. Lixin Tao



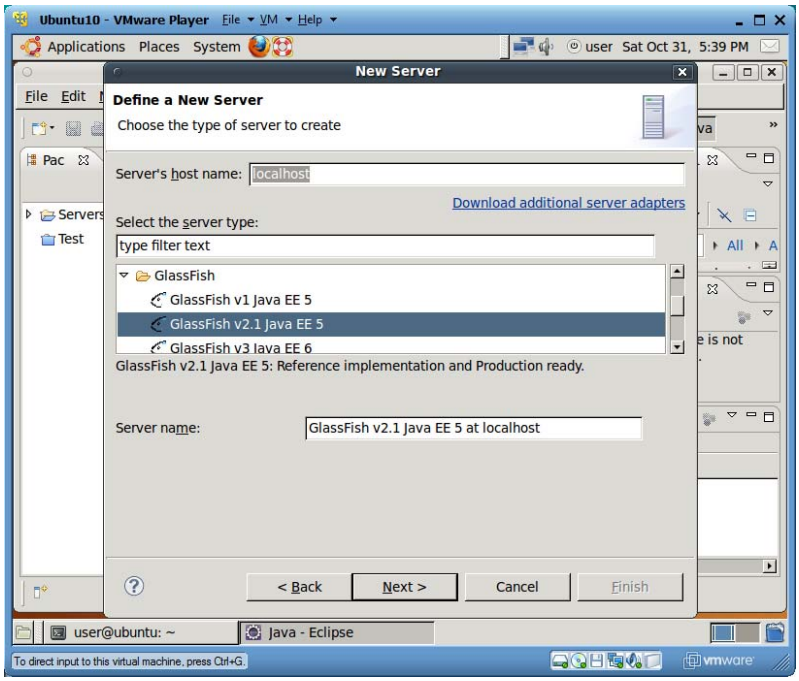

- Check the "I accept …" checkbox, and click on the *Finish* button to start downloading and installation of the *GlassFish* adapter for *Eclipse*.




- When receiving security warning, click on the *OK* button.
- When requested, restart *Eclipse* for the changes to take effect.
- Choose "File|New|Other…", and then choose *Server* in the *New* window, as shown below:

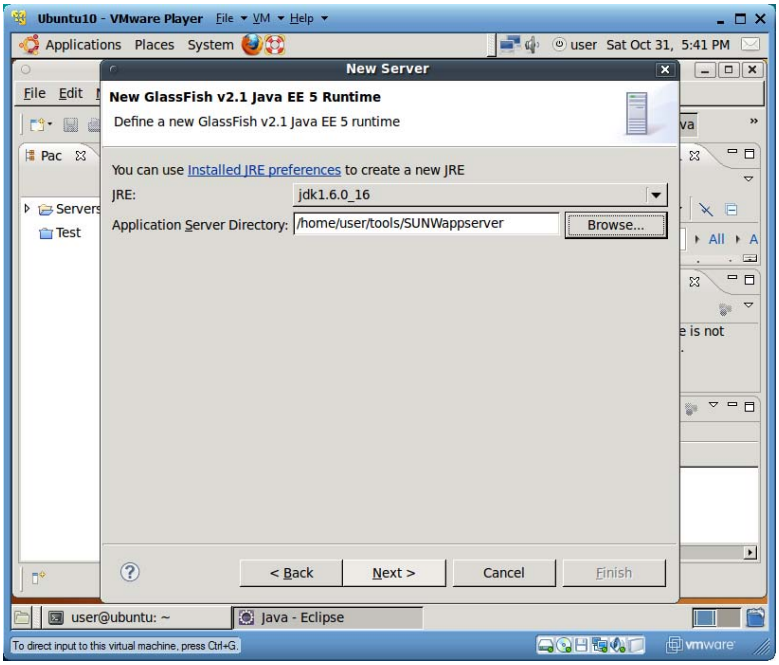



- Click on the *Next* button.

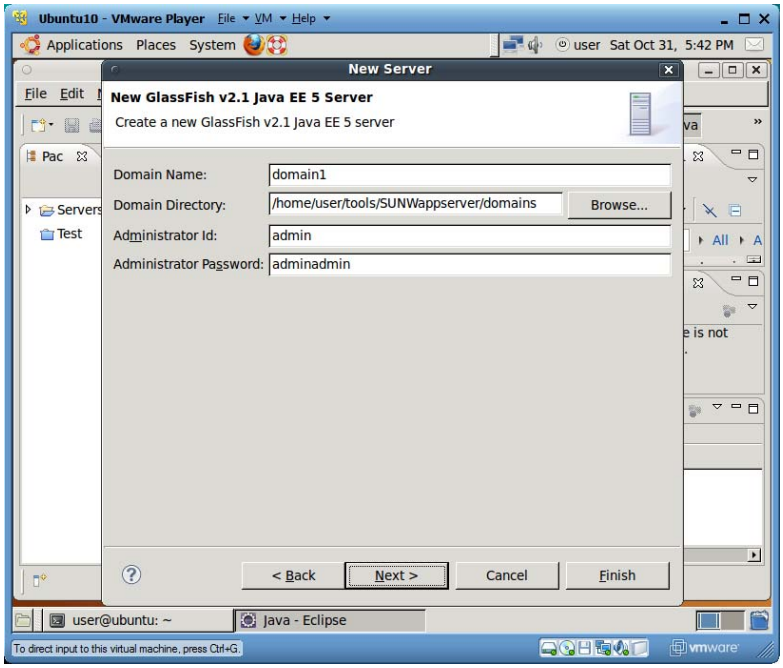



- Choose “GlassFish v2.1 Java EE 5”, and then click on the *Next* button. In the “New Server” window, choose “jdk1.6_16” for JRE, and browse for “/home/user/tools/SUNWappserver” as value for “Application Server Directory”.

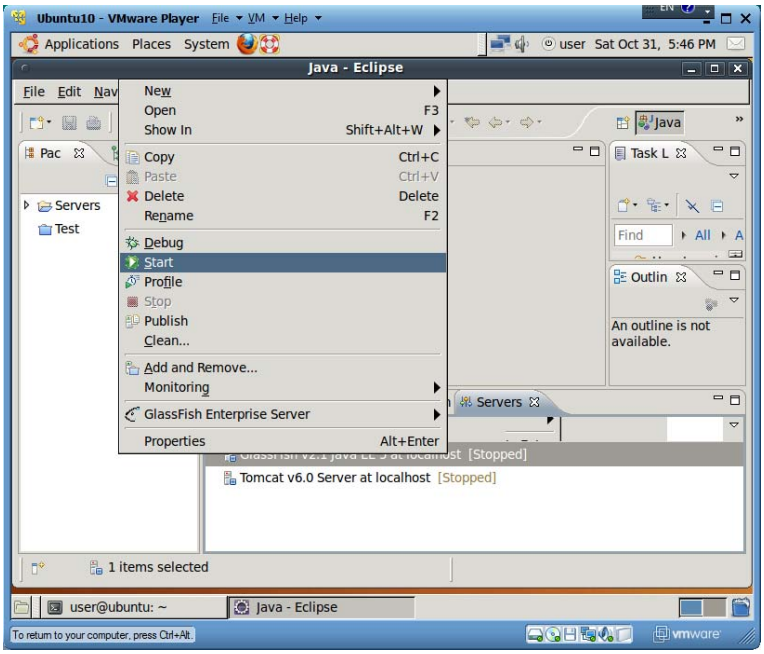
©Copyright 2009 Prof. Lixin Tao




- Click on the *Next* button.

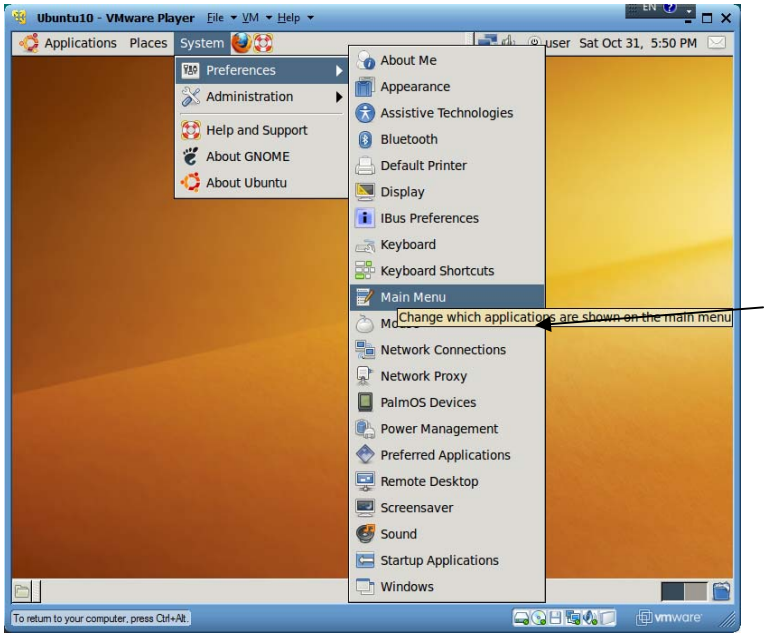



- Accept the default values, and click on the *Finish* button. Now you can use the popup menu of the *GlassFish* v2.1 application server under the *Server* tab to start/stop the *GlassFish* application server.



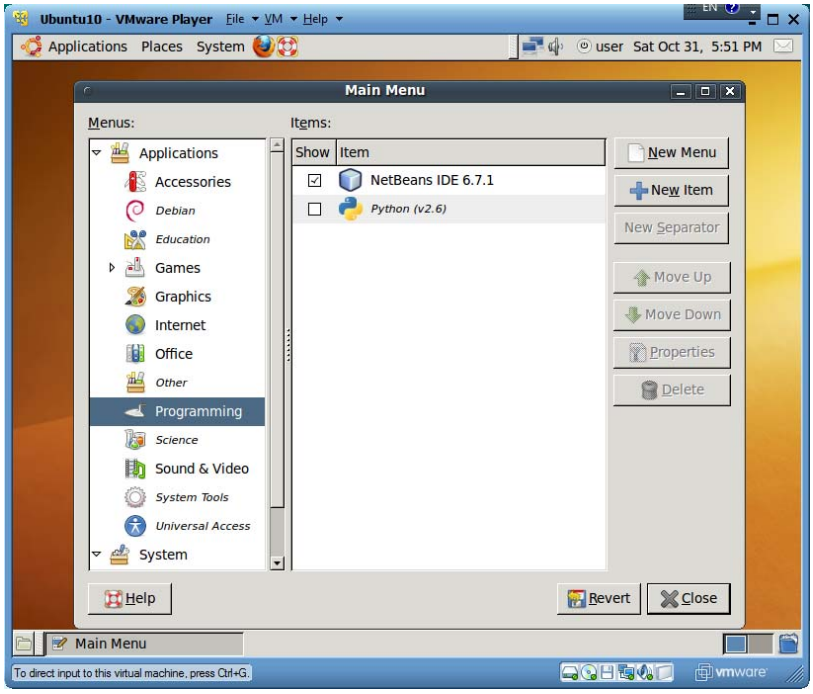

Congratulations and you have completed the installation and set up of *Eclipse* with the capacity to develop both Java projects and Java EE web applications.

As an optional step, you can add *Eclipse* to the *Ubuntu*'s "Applications|Programming" menu. To do so, click on *Ubuntu* menu item "System|Preferences|Main Menu":

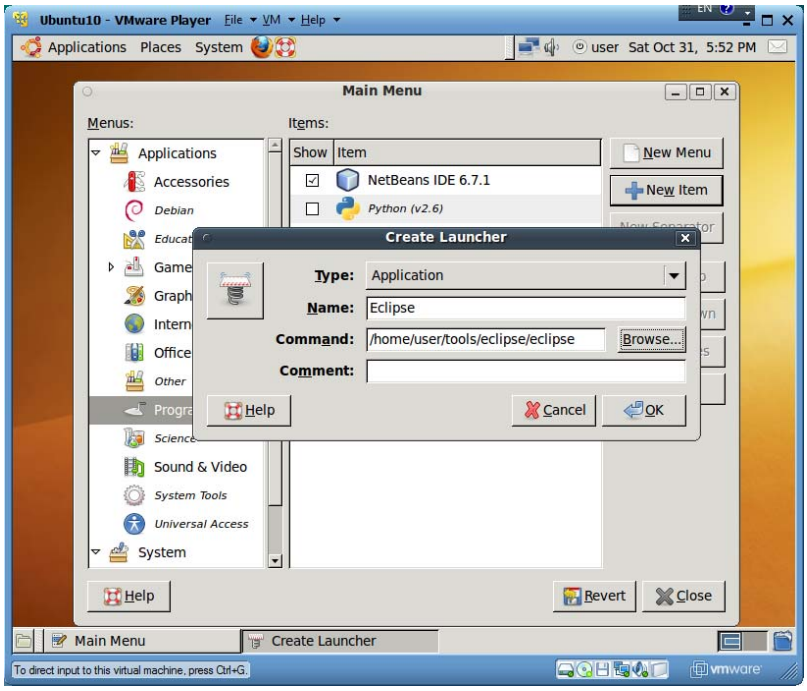



Now you see

©Copyright 2009 Prof. Lixin Tao




Choose “Programming” in Menus, and click on the “New Item” button.




Fill in Name “Eclipse” and Command “/home/user/tools/eclipse/eclipse”. Click on the *OK* button. Shut down the “Main Menu” window. Mission accomplished. Now you can use menu item “Applications|Programming|Eclipse” to launch *Eclipse*.

## 4.14 Installing Java Tutorials

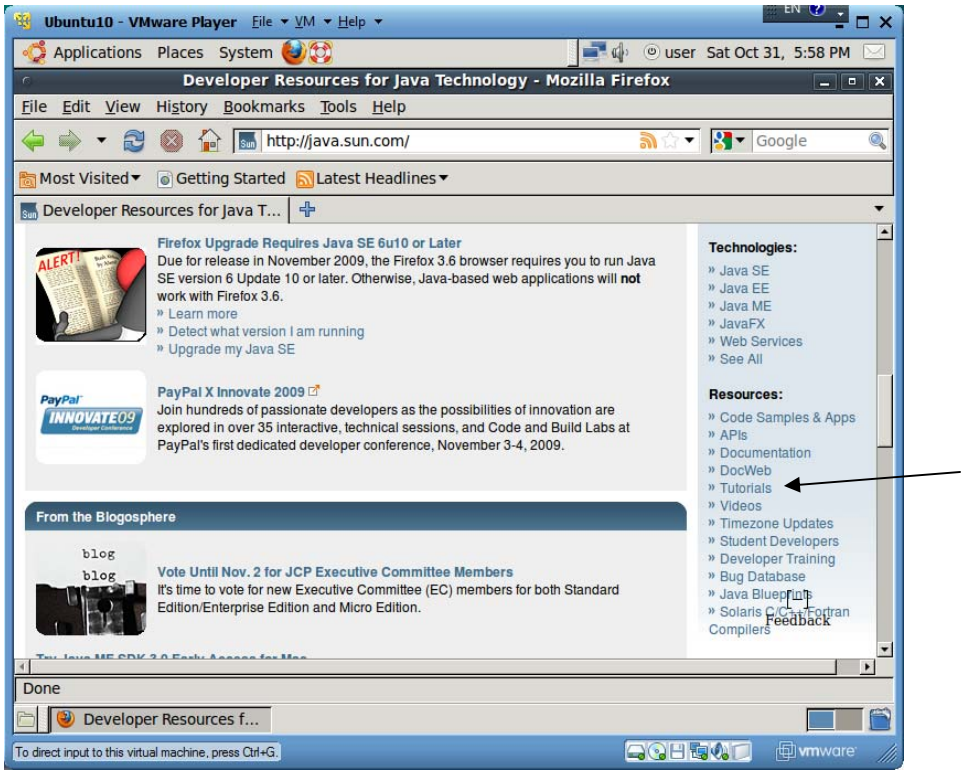
Before we can use the Java tutorials, we must have installed Java JDK.

### *4.14.1 Downloading Sun Java Tutorials*

This section shows how to download three Sun's Java tutorials: *Java tutorial* for basic client-side Java programming, *First Cup Tutorial for Java EE* for a brief taste of Java EE (Enterprise Edition), and *Java EE Tutorial* for Java EE server technologies.

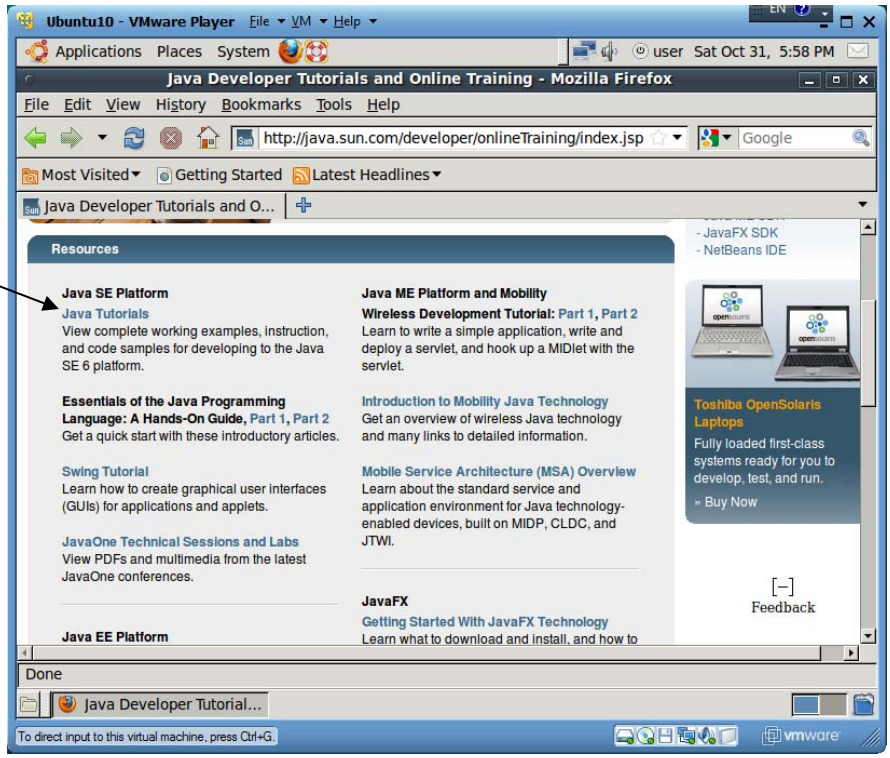
#### 4.14.1.1 Downloading *Java Tutorial*

- Use a web browser to visit http://java.sun.com

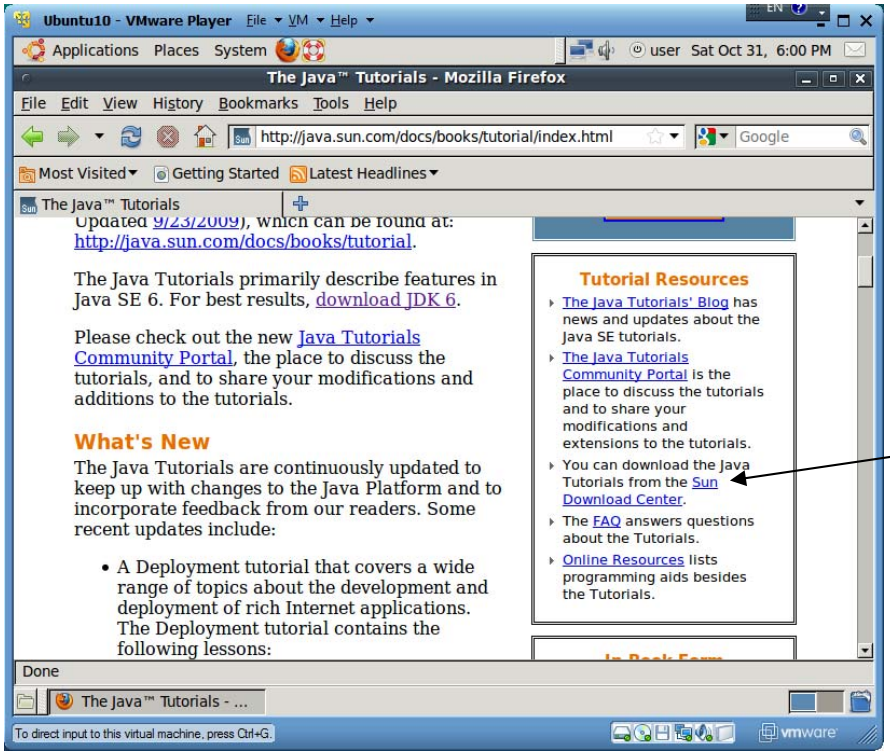



- In the *Resources* area, click on the *Tutorials* link.

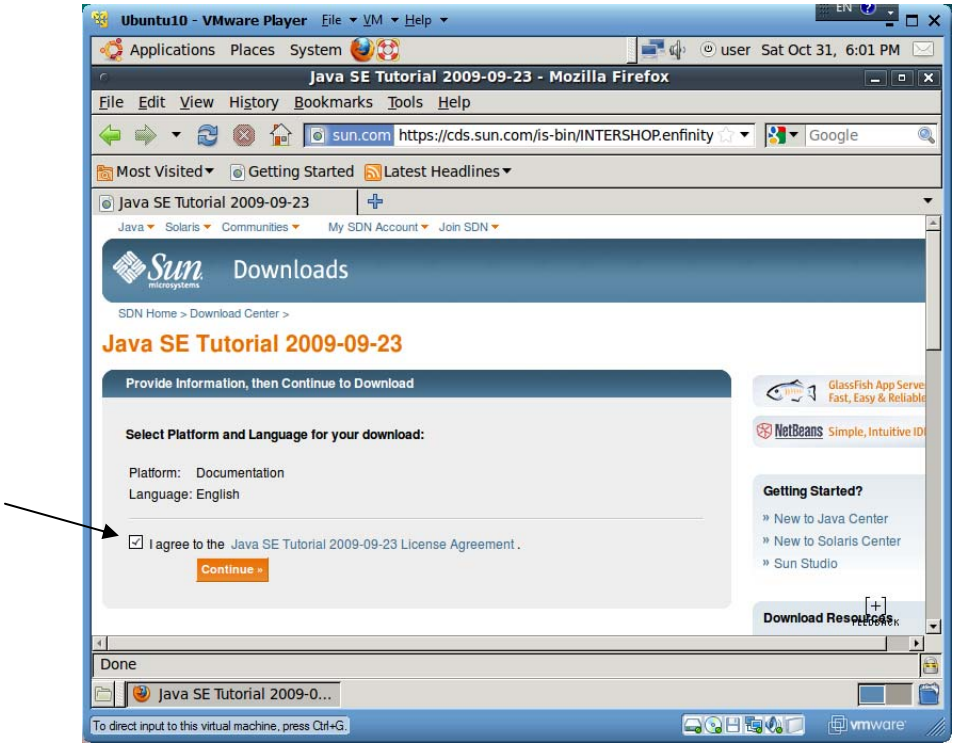
©Copyright 2009 Prof. Lixin Tao



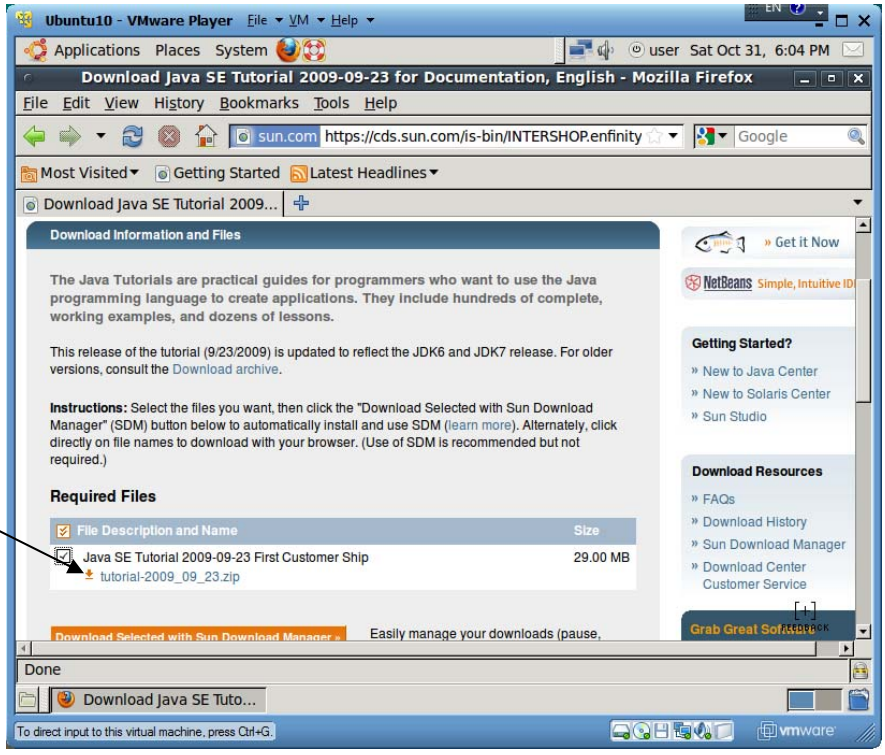

- Click on the “Java Tutorials” link under “Java SE Platform”.




- Click on the “Sun Download Center” link in the *Tutorial Resources* area.




- Check the "I agree to …" checkbox, and then click on the *Continue* button.



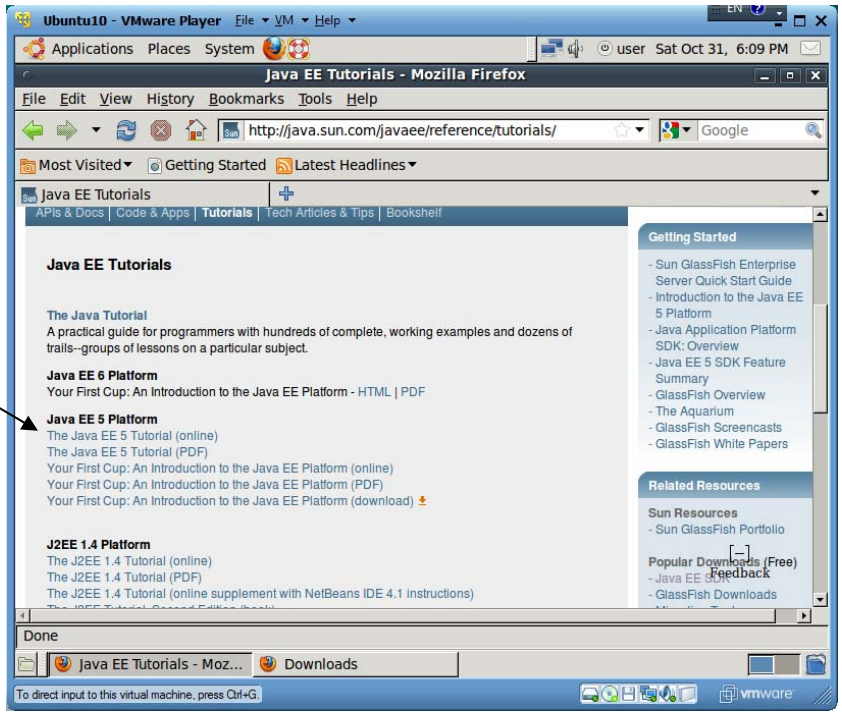

- Click on the "tutorial-2009_09_23.zip" link to download the file. Save the file to your folder *Downloads*.

©Copyright 2009 Prof. Lixin Tao

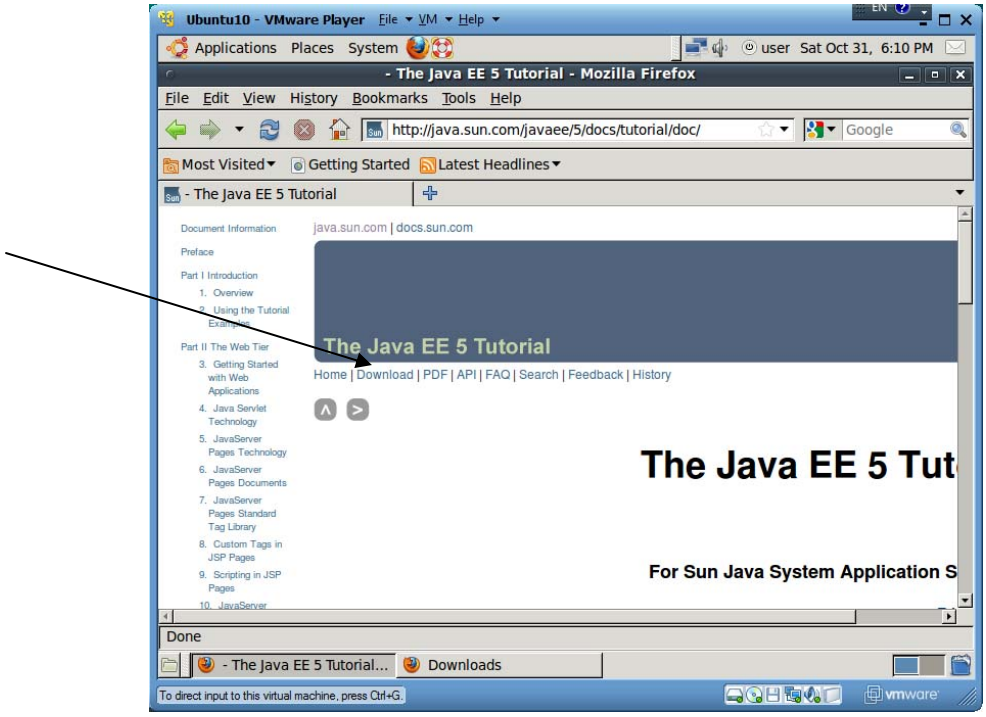
#### 4.14.1.2 Downloading *Java EE Tutorial*

- Now go back for three web pages to see the *Resources* area again.

- Click on the “Java EE Tutorial and Resources” link.

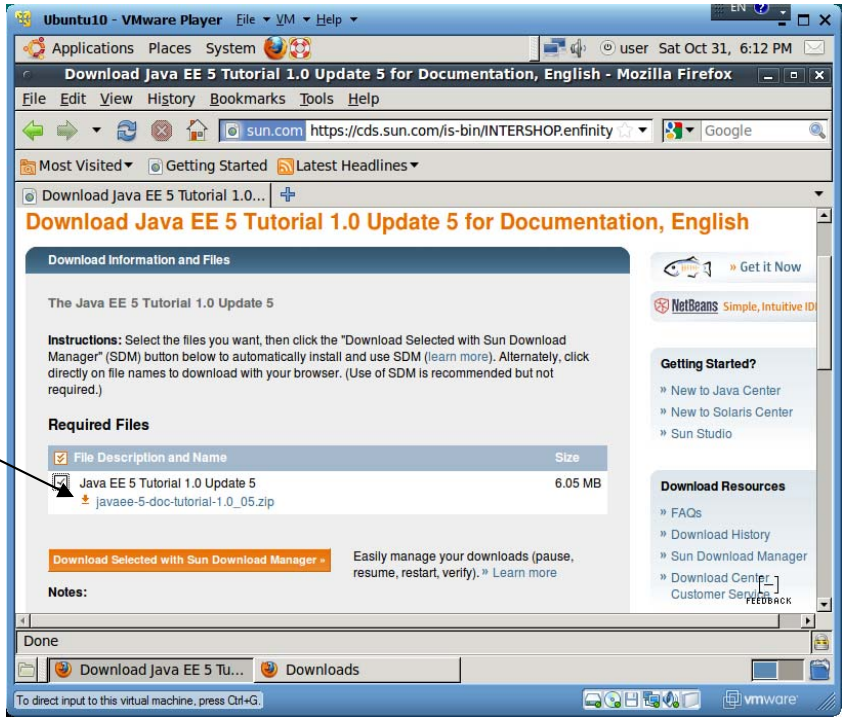



- Click on the “The Java EE 5 Tutorial (online)” link.




- Click on the “Download” link under the tutorial title.

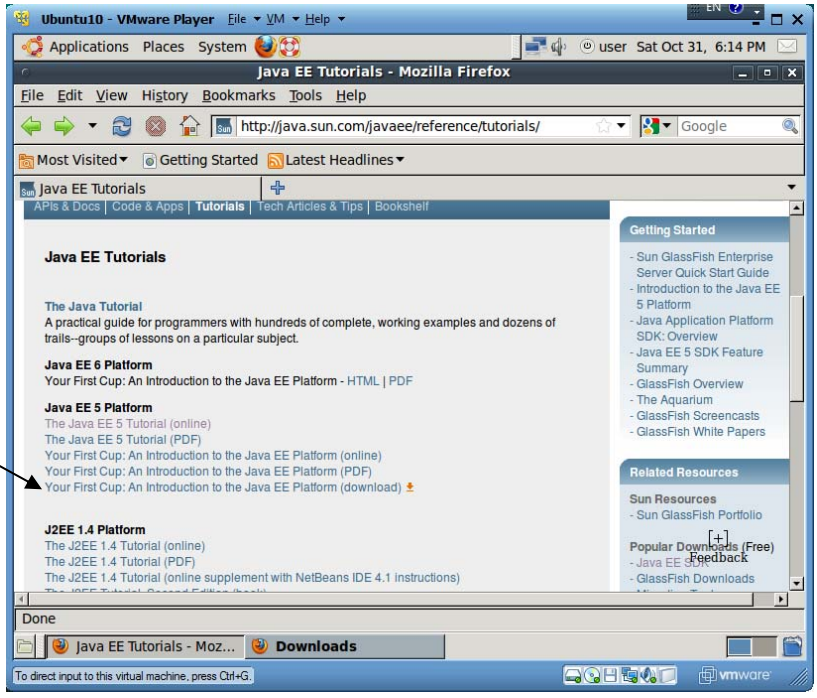
©Copyright 2009 Prof. Lixin Tao



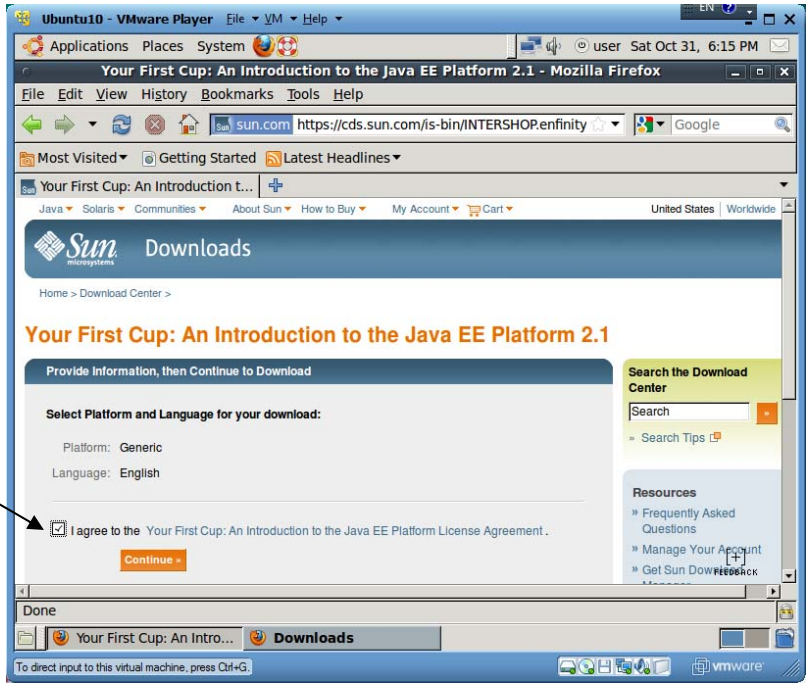

- Click on the "javaee-5-doc-tutorial-1.0_05.zip" link to download it to your folder *Downloads*.

### 4.14.1.3 Downloading *Your First Cup for Java EE* Tutorial

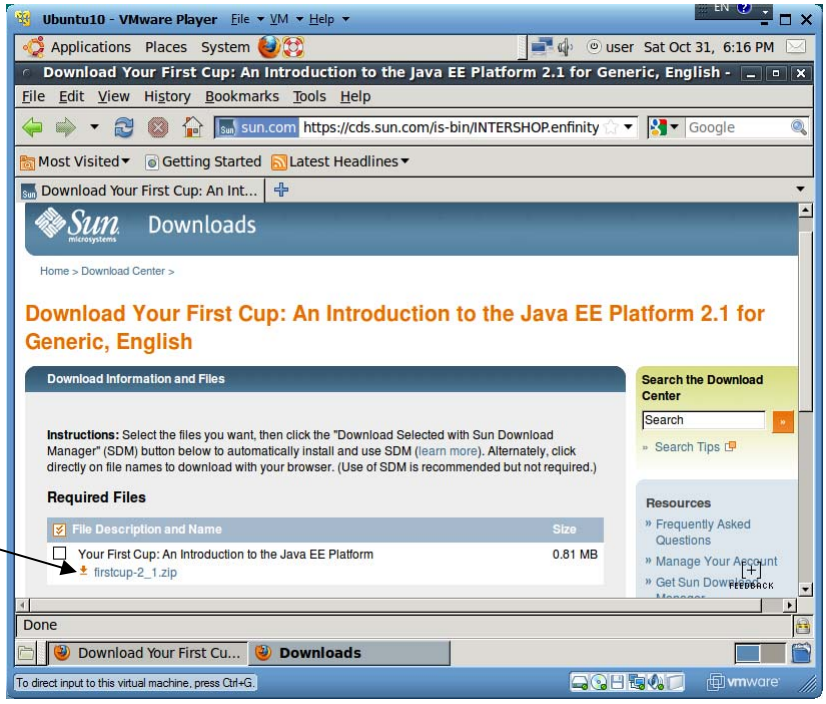
- Now go back for two pages to http://java.sun.com/javaee/reference/tutorials/.



- Click on the “Your First Cup: An Introduction to the Java EE Platform (download)” link.



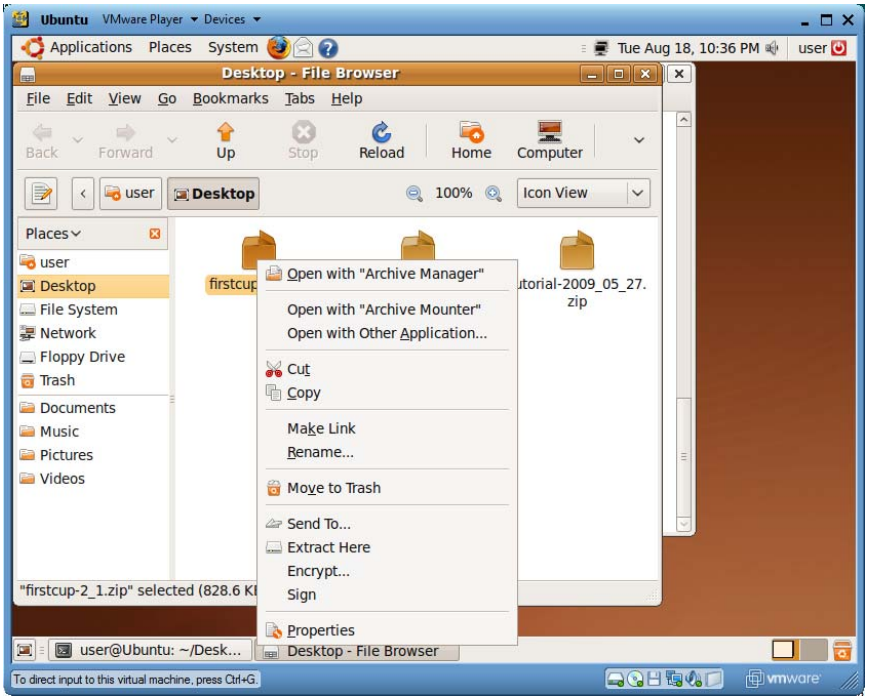

- Check the “I agree to …” checkbox, and then click on the *Continue* button.




- Click on the “firstcup-2_1.zip” link and save the file to your folder *Downloads*.

©Copyright 2009 Prof. Lixin Tao


- In a file browser, right-click on each of the three downloaded zip files and use menu item "Extract Here" to generate folders "tutorial", "firstcup", and "javaeetutorial5" in "~/Downloads".




- Run the following commands to move the three new tutorial folders to "~/JavaTutorials".

```
cd
mkdir   JavaTutorials
mv  Downloads/tutorial  JavaTutorials/JavaTutorial
mv  Downloads/firstcup  JavaTutorials/firstcup
mv  Downloads/javaeetutorial5  JavaTutorials/javaeetutorial5
```

### *4.14.2 Setting Up Java Tutorials*

To set up the Java EE tutorials, you need to install *GlassFish* application server first, which is part of our installation of the Java EE bundle for *NetBeans* and *GlassFish*.

#### 4.14.2.1 Setting up Java Tutorial

Java tutorial does not need setting up. Use a web browser to visit "/home/user/JavaTutorials/JavaTutorial/index.html" to read the tutorial.

#### 4.14.2.2 Setting Up *First Cup for Java EE* Tutorial

- Run "gedit /home/user/JavaTutorials/firstcup/example/bp-project/build.properties.sample" and set the values of the following variables as below (Note: Don't include space characters at the end of the variable values):

```
javaee.home=/home/user/tools/SUNWappserver
firstcup.tutorial.home=/home/user/JavaTutorials/firstcup
javaee.server.port=8081
AS_ADMIN_PASSWORD=adminadmin     # uncomment this line
```

- Run "gedit /home/user/JavaTutorials/firstcup/example/bp-project/admin-password.txt" and make its contents to contain only the following line:

```
AS_ADMIN_PASSWORD=adminadmin
```

Before we can use the *First Cup for Java EE* Tutorial, we must have installed Java EE 5 JDK which is part of our earlier *NetBeans* installation.

#### 4.14.2.3 Setting Up Java EE 5 Tutorial

Before we can use the Java EE 5 Tutorial, we must have installed Java EE 5 JDK which is part of our earlier *NetBeans* installation.

- Run "gedit /home/user/JavaTutorials/javaeetutorial5/examples/bp-project/build.properties.sample" and set the values of the following variables as below (Note: Don't include space characters at the end of the variable values):

```
javaee.home=/home/user/tools/SUNWappserver
javaee.tutorial.home=/home/user/JavaTutorials/javaeetutorial5
javaee.server.port=8081
AS_ADMIN_PASSWORD=adminadmin     # uncomment this line
```

- Run "cd /home/user/JavaTutorials/javaeetutorial5/examples/bp-project" to change working folder. Run "cp build.properties.sample build.properties" to make a copy of file "build.properties.sample" and name the copy "build.properties".
- Run "gedit /home/user/JavaTutorials/javaeetutorial5/examples/common/admin-password.txt" and make its contents to contain only the following line:

```
AS_ADMIN_PASSWORD=adminadmin
```

#### 4.14.2.4 Populating Books in Database for Java EE Bookstores

Several bookstore web applications of the *Java EE Tutorial* share the same *Derby* database tables for book information. To populate the database tables,

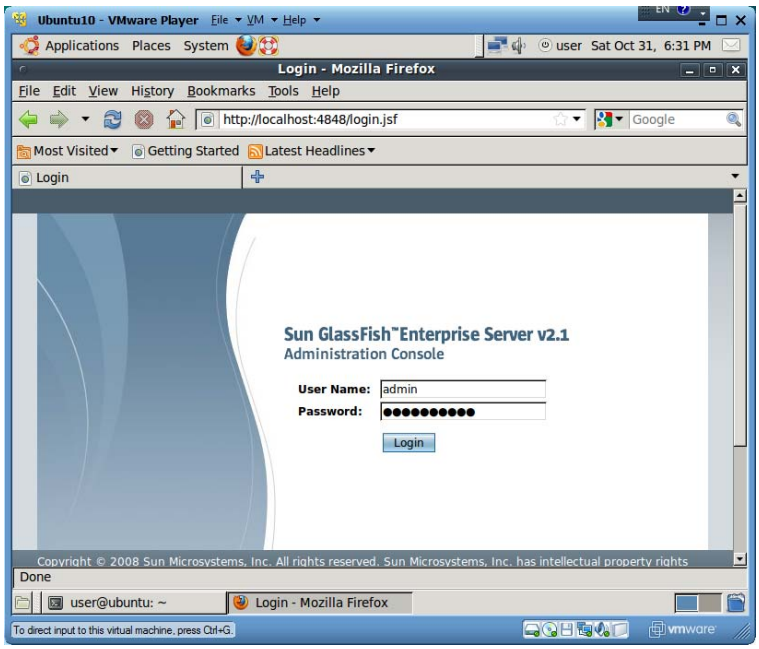
©Copyright 2009 Prof. Lixin Tao

- Run “cd /home/user/JavaTutorials/javaeetutorial5/examples/web/books” to change the working folder.
- Run “ant create-tables” to populate the books database tables.

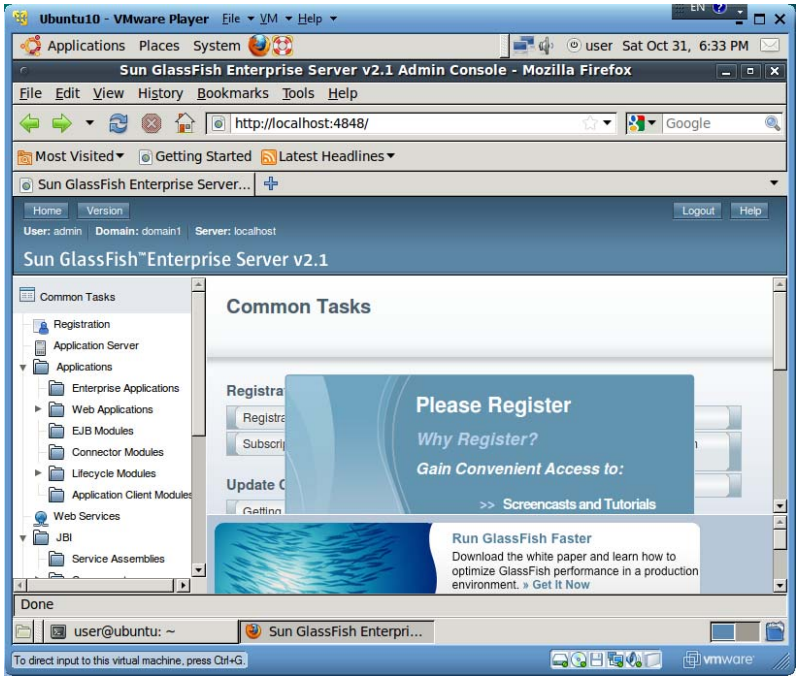
#### 4.14.2.5 Defining Java EE Bookstores’ Data Source

Several bookstore web applications of the *Java EE Tutorial* use the logical JNDI name “jdbc/BookDB” to access their *Derby* database tables. Here we define JNDI name “jdbc/BookDB”.

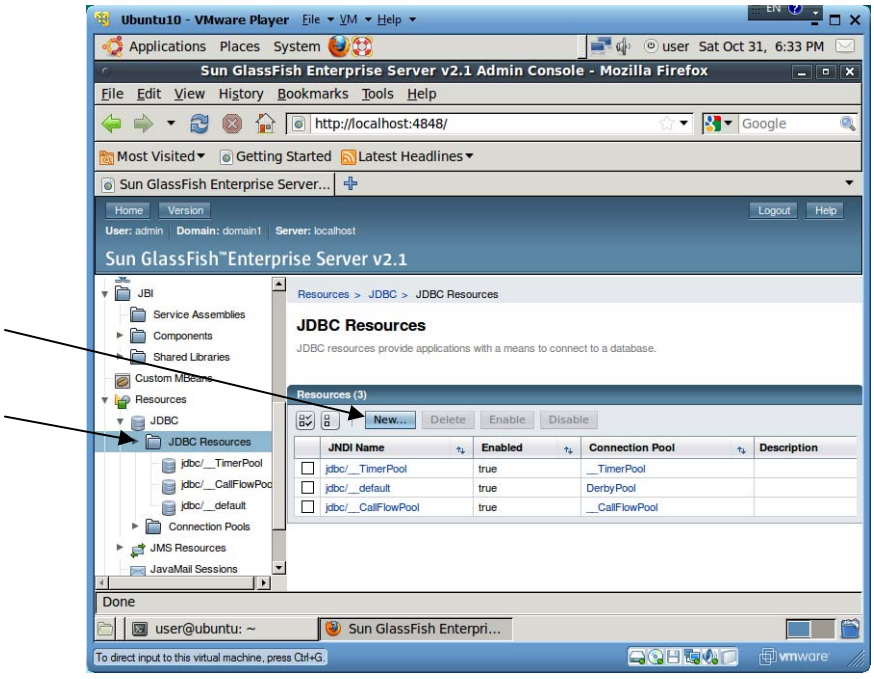
- Run “**asadmin start-domain --verbose domain1**”, or “glassfish-start”, to launch *GlassFish* application server.
- Use *Firefox* web browser to visit http://localhost:4848 to access *GlassFish* admin console. Enter “admin” for *User Name*, and “adminadmin” for *Password*.




- Click on the *Login* button.

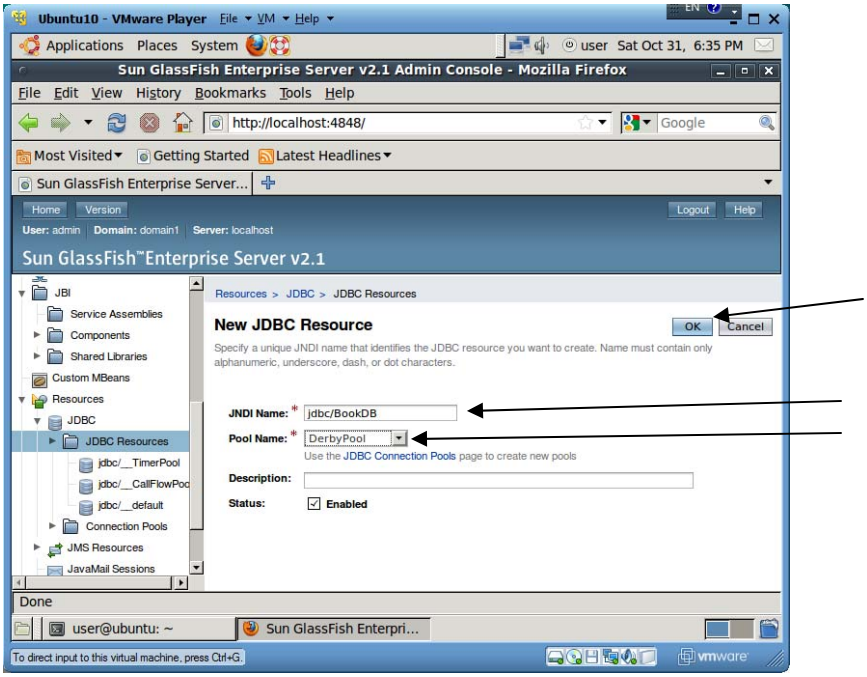



- Click to expand the *Resources* node in the left "Common Tasks" pane. Click to expand the nested *JDBC* node. Select the nested "JDBC Resources" node.

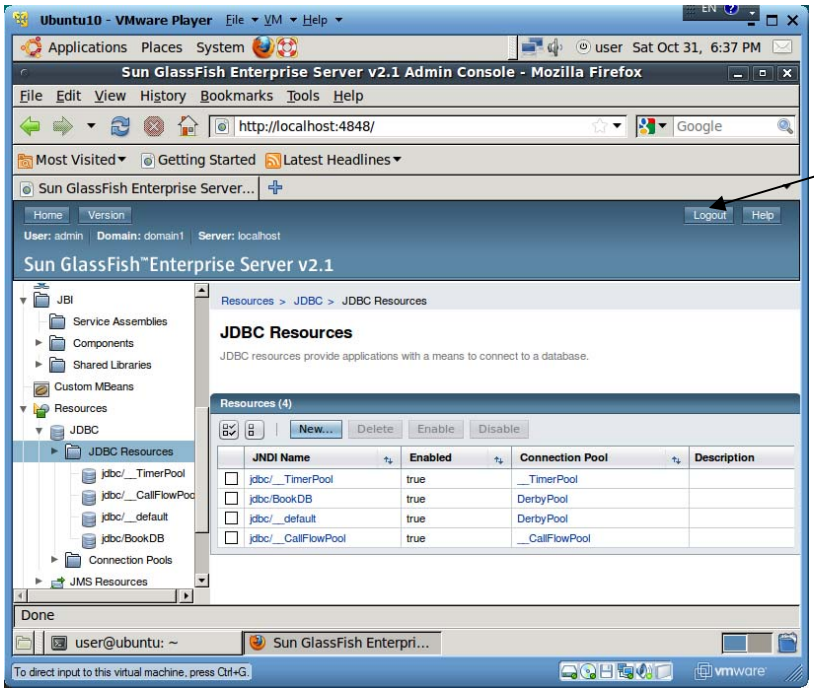



- Click the "New…" button on top of the *Resources* table in the right pane. Type "jdbc/BookDB" in the *JNDI Name* field, and choose "DerbyPool" in the *Pool Name* combo box.

©Copyright 2009 Prof. Lixin Tao




- Click the *OK* button to complete the process.



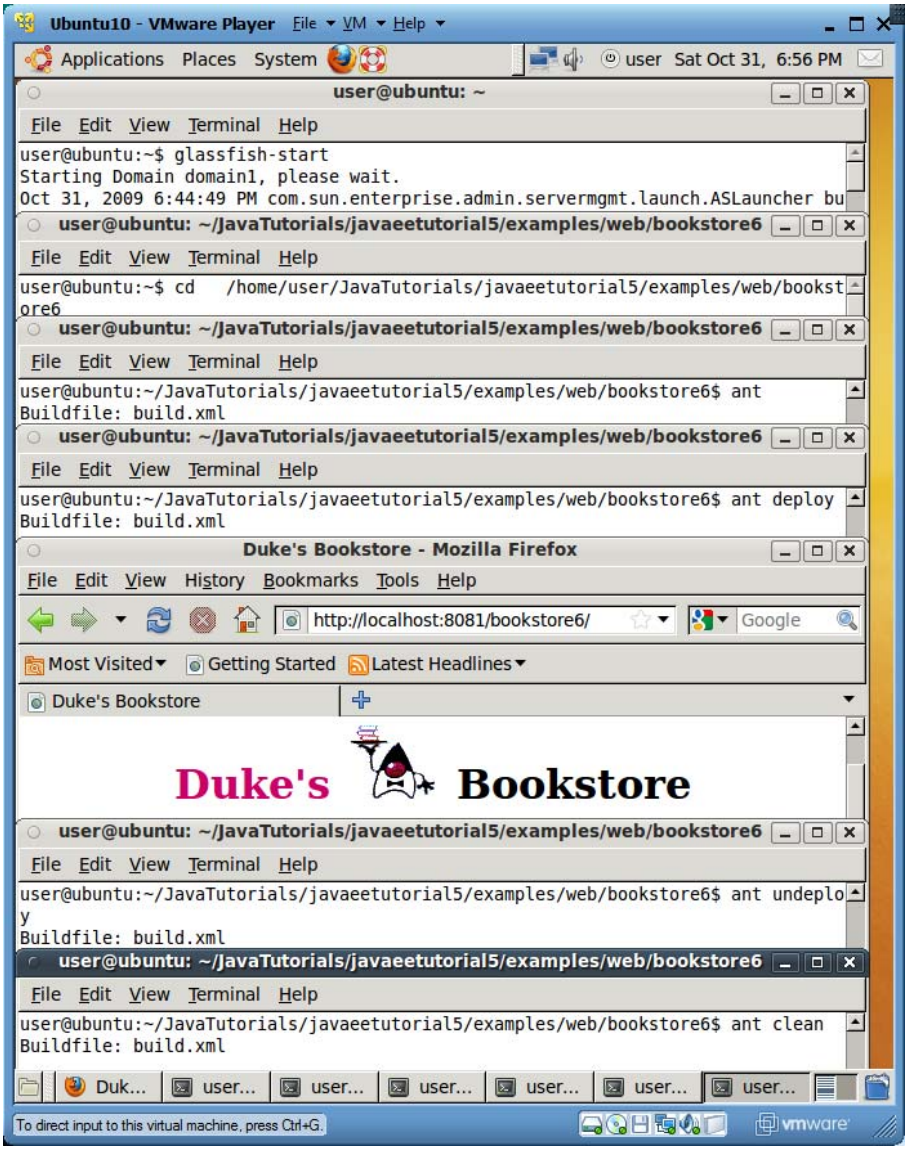

- Now you can logout from the admin console.

### 4.14.2.6 Validating *GlassFish* and *Java EE Tutorial* Setting Up

Now we run an example web application, *bookstore6*, of the *Java EE Tutorial* to validate our *GlassFish* and *Java EE Tutorial* setting up.

- Start a terminal window, and run "glassfish-start" to launch the *GlassFish* application server.
- Run "cd /home/user/JavaTutorials/javaeetutorial5/examples/web/bookstore6" to change working folder to where *bookstore6* is located.
- Run "ant" to build the web application and generate a WAR (Web Archive) file for it. Utility "ant" is a Java version of "make" on Unix/Linux and "nmake" on Windows. When you run "ant", it will find a project configuration file "build.xml" in the current folder to see how to carry out your command.
- Run "ant deploy" to deploy project *bookstore6* to the *GlassFish* application server. This step will also implicitly launch the *Derby* database server.
- Use *Firefox* web browser to visit http://localhost:8081/bookstore6. If you can see the web page for "Duke's Bookstore", congratulations and you have succeeded. Otherwise you need to redo instructions in this section precisely.
- Run "ant undeploy" to undeploy web application *bookstore6*. If you reload your web browser at http://localhost:8081/bookstore6 you will find the web page is not available any more.
- Run "ant clean" to delete all files generated in the build step.
- Run "glassfish-stop" to shut down the *GlassFish* application server.

©Copyright 2009 Prof. Lixin Tao




## 4.15 Deploying Example Web Applications on Apache

### *4.15.1 Installing Drupal*

*Drupal* is one of the most popular open-source content management platforms for developing advanced web applications for social networking, blogging and conference management. This subsection shows you how to install *Drupal* on our *Apache* web server.

#### 4.15.1.1 Enabling MOD_REWRITE and .htaccess on *Apache*

- Run command “cd /etc/apache2/mods-available” to change working folder to the folder holding all available *Apache* modules. Run command “sudo a2enmod rewrite” to enable module rewrite on *Apache*.
- Run “sudo gedit /etc/apache2/sites-enabled/000-default” and replace

```
        <Directory />
              Options FollowSymLinks
              AllowOverride None
        </Directory>
        <Directory /var/www/>
              Options Indexes FollowSymLinks MultiViews
              AllowOverride None
              Order allow,deny
              allow from all
        </Directory>
```

with

```
        <Directory />
              Options FollowSymLinks
              AllowOverride All
        </Directory>
        <Directory /var/www/>
              Options Indexes FollowSymLinks MultiViews
              AllowOverride All
              Order allow,deny
              allow from all
        </Directory>
```

(Changing the value of "AllowOverride" from *None* to *All*)

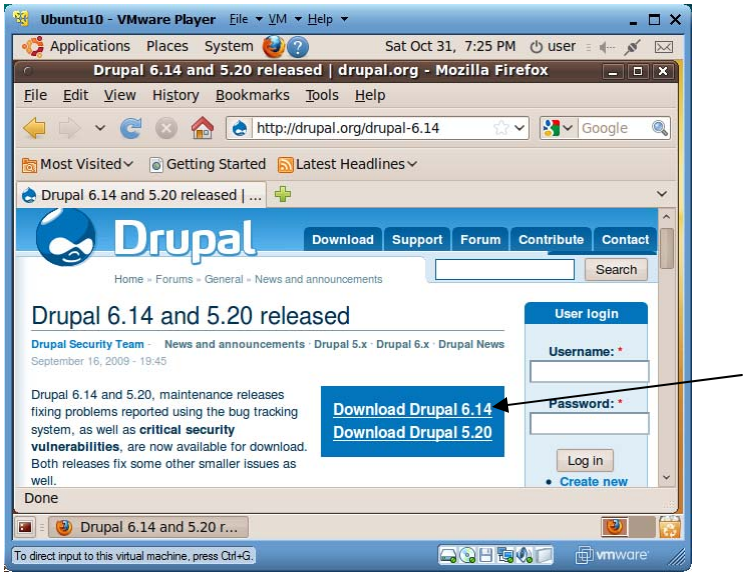
- Run command "sudo apache2ctl restart" (or "sudo /etc/init.d/apache2 restart") to restart and activate the new configuration.

#### 4.15.1.2 Installing *Drupal*

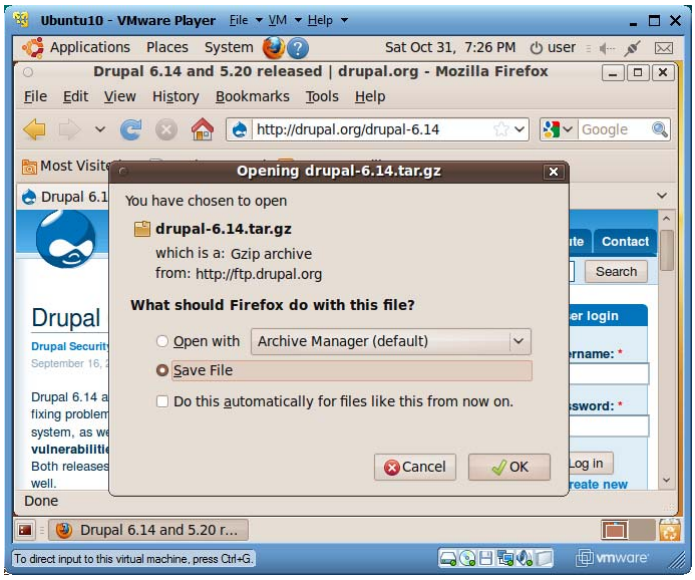
- Launch *Firefox* to visit http://drupal.org

©Copyright 2009 Prof. Lixin Tao

- Click on the “Drupal 6.14” (or other latest version) link in the *Download* pane.




- Click on the “Download Drupal 6.14” link to start downloading.




- Check the “Save File” checkbox, and click on the *OK* button to start the downloading of file “drupe-6.14.tar.gz” into “~/Downloads”.
- Run “cd ~/Downloads” to change working folder.
- Run “sudo tar xvzf drupal-6.14.tar.gz -C /var/www” to extract the installer contents into *Apache* document root folder “/var/www” (~/www). A new folder “drupal-6.14” is created.
- Run “rm drupal-6.14.tar.gz” to delete the *Drupal* installer.
- Run “cd /var/www” to change working folder.
- Run “sudo mv drupal-6.14 drupal” to rename folder “drupal-6.14” to “drupal”.
- Run “cd /var/www/drupal/sites/default” to change working folder.

- Run “sudo cp default.settings.php settings.php” to make a new file “settings.php”.
- Run “sudo mkdir files” to create a new folder “files” for holding *Drupal* web site files.
- Run “sudo chmod a+w files” to allow anyone to create files in folder “files”.
- Run “sudo chmod a+w settings.php” to make file “settings.php” modifiable by all users.
- Run “cd ..” to move up one level of working folder to “/var/www/drupal/sites”.
- Run “sudo chmod a+w default” so files can be generated in folder “default”.
- Run “sudo gedit /var/www/drupal/.htaccess” to uncomment (removing the leading #) the following line:

```
RewriteBase /drupal
```

#### 4.15.1.3 Installing PHP5-GD and POSTFIX

These steps install *GD* image library support for *PHP5* and the *Postfix* module for your Linux PC to send emails.

- Run “sudo aptitude install php5-gd” to install the module supporting *GD* image library in *PHP5*.
- Run “sudo aptitude install postfix” to allow your Linux PC to send emails. After reading the explanations for *Postfix* configuration, type Tab key to access the “Postfix Configuration” window. Accept the default selection “Internet Site”, and use Tab key to select the *OK* button, then type *Enter* key. Take the default values to complete the *Postfix* installation.

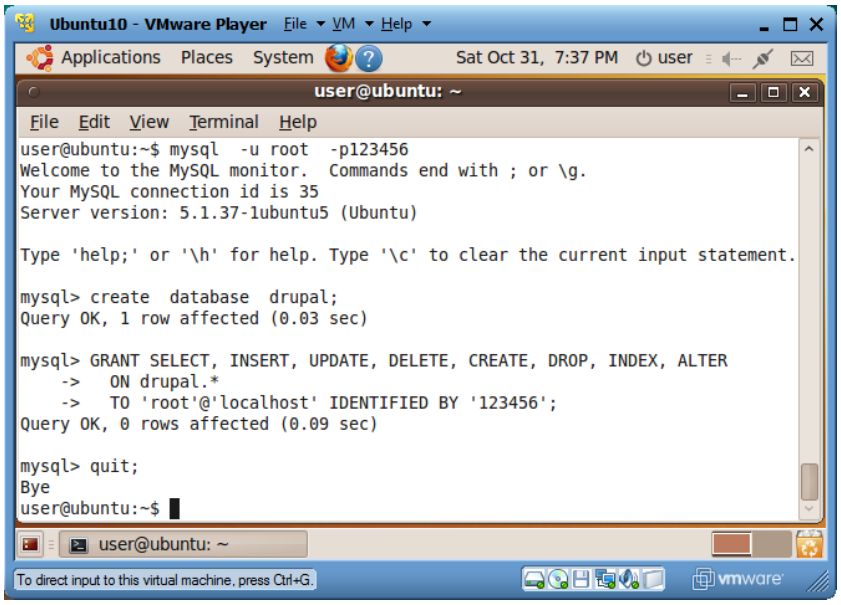
#### 4.15.1.4 Creating a *MySQL* database “drupal”

- Run “mysql -u root -p123456” to login *MySQL* admin console.
- Run “create database drupal;” to create database “drupal”.
- Run the following command to let MySQL root manipulate database “drupal”:

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER
  ON drupal.*
  TO 'root'@'localhost' IDENTIFIED BY '123456';
```

- Run “quit;” to quit the *MySQL* admin console.

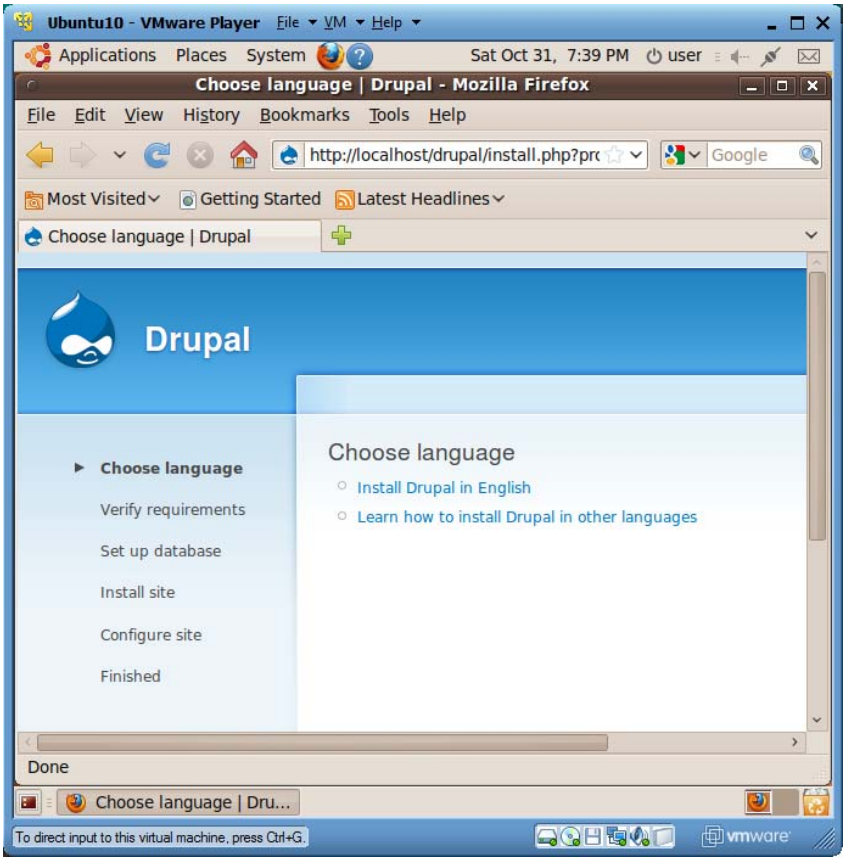
©Copyright 2009 Prof. Lixin Tao



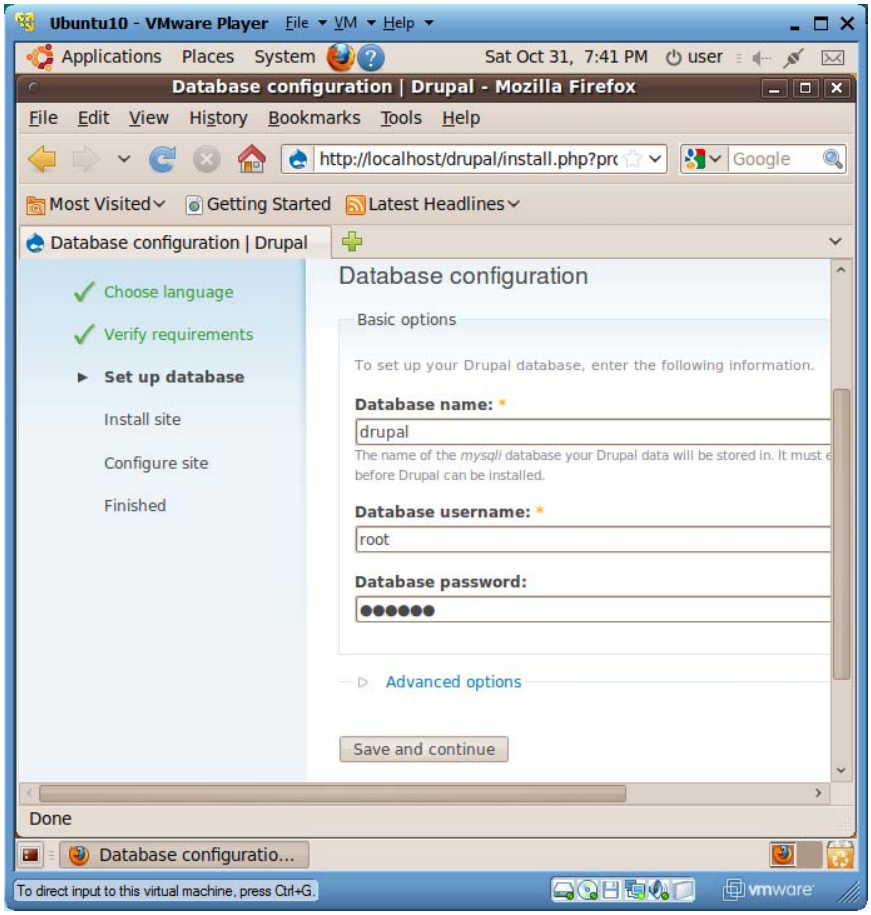

- Run “sudo aptitude install php5-mysql” to install MySQL support in *PHP5*.
- Run “sudo apache2ctl restart” to restart *Apache* and enable the *MySQL* support in *PHP5*.

4.15.1.5 Configuring *Drupal*

- Launch *Firefox* web browser to visit http://localhost/drupal and you will see the following screen:

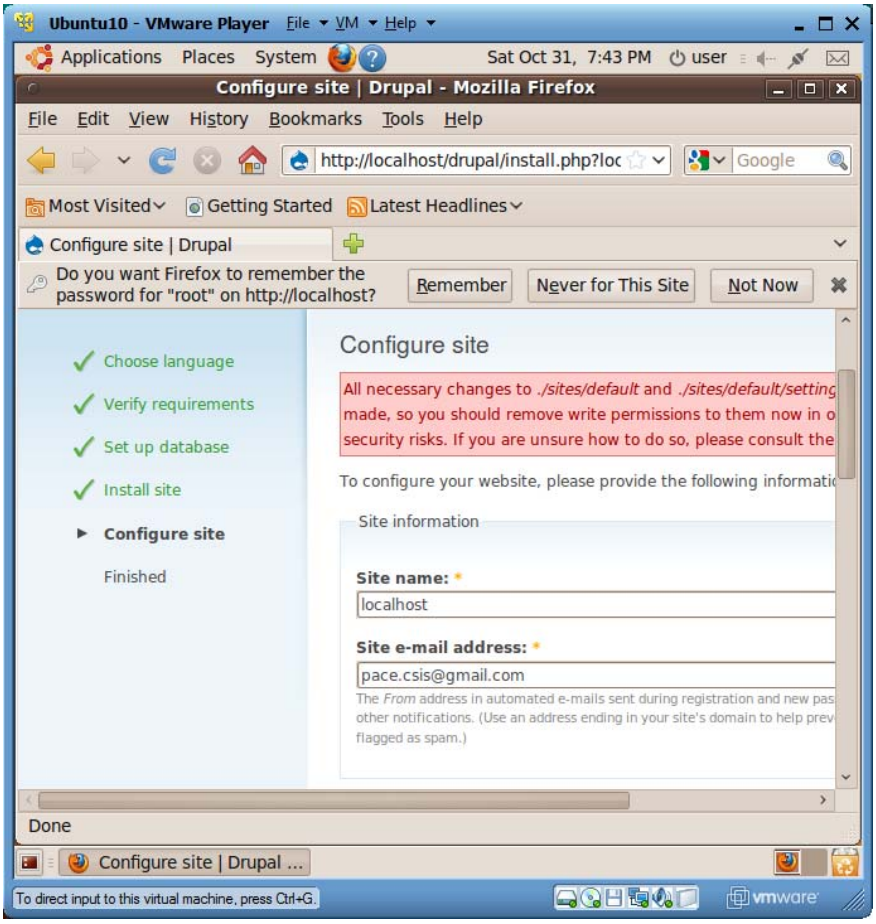


- Click on link “Install Drupal in English” to see the following screen.

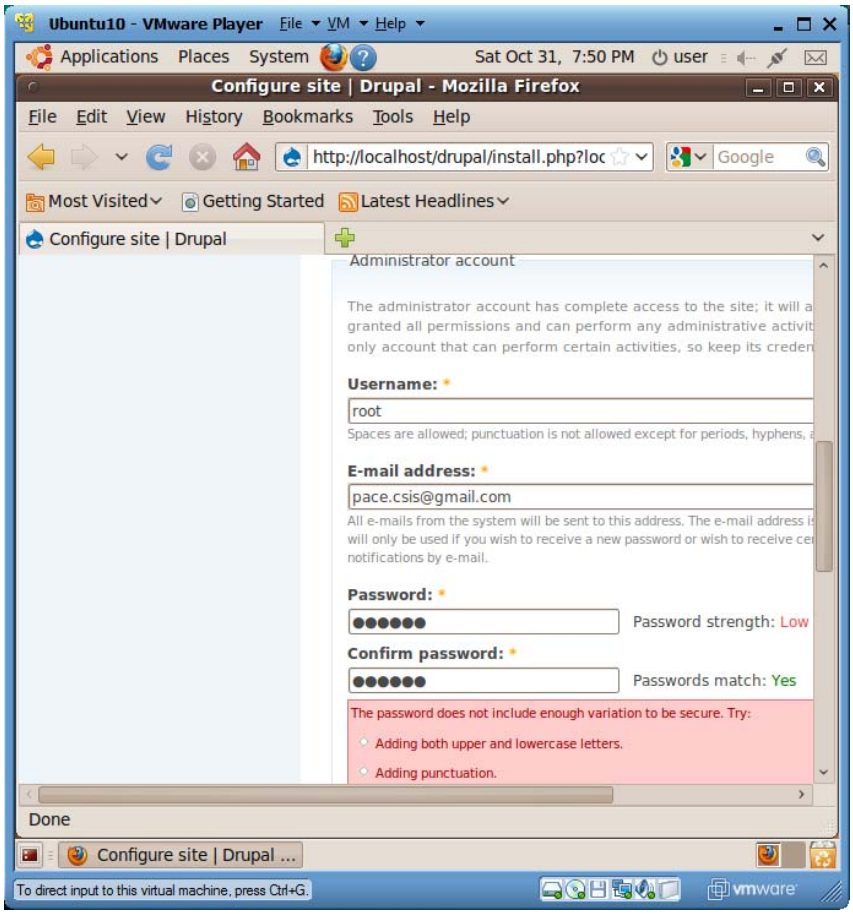



- In the “Database name” field, enter “drupal”. In the “Database username” field, enter “root”. In the “Database password” field, enter “123456”. Click on “Save and continue” to go to the next step.
- Here we enter “Site information”. In the “Site name” field, keep the default “localhost”. In the “Site e-mail address” field, enter a valid email address of yours like “pace.csis@gmail.com”. The following is the first half of the “Configure site” web page.

©Copyright 2009 Prof. Lixin Tao

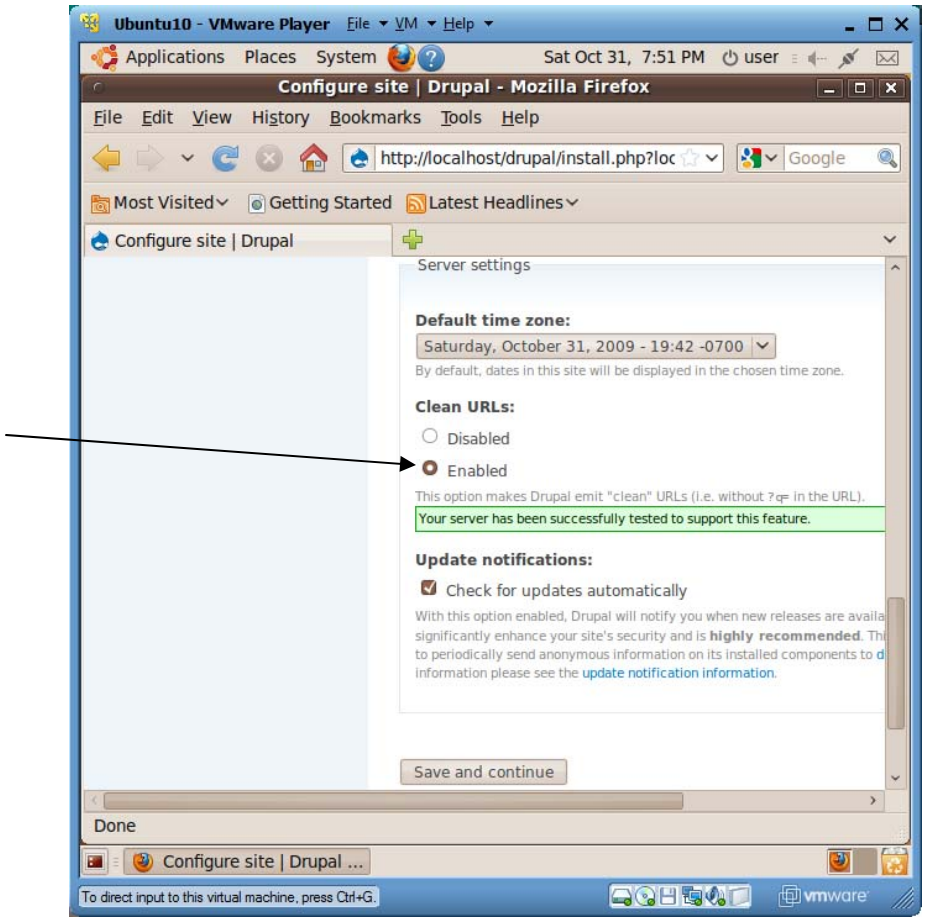



- Next we enter information for the “Administrator account” panel. In the “Username” field enter “root”. This is *Drupal*’s admin account. In the “E-mail address” field, enter the same email address that you entered in the “Site information” panel (it should be filled in automatically). In the “Password” and “Confirm password” fields, enter “123456”, or more secure password that you can remember. I used 123456 so you can easily find all passwords in our test installations.

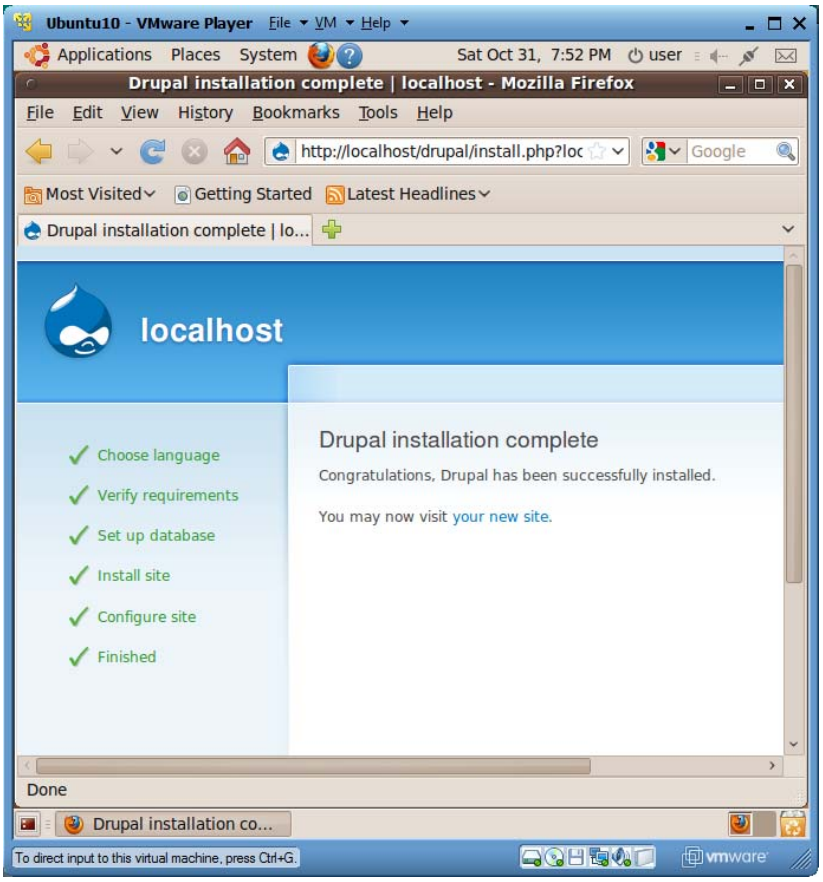



- Here we enter information for the “Server settings” panel. Check the “Enabled” checkbox for “Clean URLs”.

©Copyright 2009 Prof. Lixin Tao




- Click on “Save and continue”.




#### 4.15.1.6 Securing *Drupal*

- Run command "sudo chmod a-w /var/www/drupal/sites/default/settings.php" to make file "settings.php" not modifiable.
- Run command "sudo chmod a-w /var/www/drupal/sites/default" to make folder "default" not modifiable.

### *4.15.2 Installing Sample Tomcat Web Applications*

This sub-section shows how to download and deploy a set of sample web applications of mine for you to deploy in our *Tomcat* web server, and how to let *Apache* delegate HTTP requests for these applications to *Tomcat*. These sample web applications are frequently used in our future tutorials on web architectures and how to develop and secure Java web applications.

- Use *Firefox* to visit http://csis.pace.edu/lixin/download/TomcatWebApps.7z and save the file to your Linux folder "~/Downloads".
- In file browser, right click on file "TomcatWebApps.7z", and click on menu item "Extract Here" to create folder "~/Downloads/TomcatWebApps".
- Run "cd ~/Downloads" to change the working folder to *Downloads*.
- Run "sudo chmod -R a+rw TomcatWebApps" to recursively make folder "TomcatWebApps" and all files and folders under "TomcatWebApps" readable and modifiable by all users on this Linux PC.

©Copyright 2009 Prof. Lixin Tao

- Run "sudo mv ~/Downloads/TomcatWebApps ~/tomcat/webapps" to move the new folder to "webapps/TomcatWebApps".
- Run "cd ~/tomcat/webapps" to change working folder.
- Run "cp TomcatWebApps/*.war ." to copy all files with file name extension ".war" (Web Archive) from folder "TomcatWebApps" one level up to "webapps".
- Run commands

```
sudo cp -r ~/tomcat/webapps/TomcatWebApps/endorsed ~/tools/jdk1.6.0_16/jre/lib
sudo chmod -R 777 ~/tools/jdk1.6.0_16/jre/lib/endorsed
```

  to copy folder "endorsed". All jar files in folder "~/tools/jdk1.6.0_16/jre/lib/endorsed" are automatically accessible by Java programs during their compilation or execution. This step is mainly for supporting the development of *Axis* web service applications as well as compiling Java servlets in terminal windows.
- If *Tomcat* is not running, start it by running command "tomcat-start". You will notice that one new folder is created under "webapps" for each WAR file under "webapps". This is the automatic web application deployment.
- Run "tomcat-stop" to shut down *Tomcat*.
- Run "rm ~/tomcat/webapps/*.war" to delete all WAR files in folder "webapps". If you have not correctly shut down *Tomcat* before you delete the WAR files, the web applications folders generated earlier from these WAR files will be deleted too.
- From now on you can safely modify my example web applications.

#### 4.15.2.1 Creating *MySQL* Database Tables for *Tomcat* Sample Web Applications

- Now we need to create a database table for the two *bookstore* sample projects.
  - Open terminal window and run "cd ~/tomcat/webapps/bookstoreServlet" to change the working folder.
  - Run "mysql -u root -p123456" to log in *MySQL* admin console.
  - Run "use test;" after the "mysql>" prompt to change the current database to "test".
  - Run "source books.sql;" after the "mysql>" prompt to create a table and populate sample book data in the table in database "test".
  - Run "quit;" after the "mysql>" prompt to quit the *MySQL* admin console.

The following is copied from my terminal window when I ran the above steps.

```
user@ubuntu:~/tomcat/webapps/bookstoreServlet$ mysql -u root -p123456
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 57
Server version: 5.1.37-1ubuntu5 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> use test;
Database changed
mysql> source books.sql;
Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.08 sec)
```

```
   ……
   mysql> quit;
   Bye
   user@ubuntu:~/tomcat/webapps/bookstoreServlet$
```

- Now we need to create a database table for the *survey* sample projects
  - Open terminal window and run "cd ~/tomcat/webapps/survey" to change the working folder.
  - Run "mysql -u root -p123456" to login *MySQL* admin console.
  - Run "use test;" after the "mysql>" prompt to change the current database to "test".
  - Run "source survey.sql;" after the "mysql>" prompt to create an empty table for survey data in database "test".
  - Run "quit;" after the "mysql>" prompt to quit the *MySQL* admin console.

The following is copied from my terminal window when I ran the above steps.

```
   user@ubuntu:~/tomcat/webapps/survey$ mysql  -u root  -p123456
   Welcome to the MySQL monitor.  Commands end with ; or \g.
   Your MySQL connection id is 57
   Server version: 5.1.37-1ubuntu5 (Ubuntu)

   Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

   mysql> use test;
   Database changed
   mysql> source survey.sql;
   Query OK, 0 rows affected, 1 warning (0.00 sec)

   Query OK, 0 rows affected (0.01 sec)

   mysql> quit;
   Bye
   user@ubuntu:~/tomcat/webapps/survey$
```

#### 4.15.2.2 Letting *Apache* Delegate Sample *Tomcat* Web Applications

- Run "sudo gedit /etc/apache2/sites-enabled/000-default" to edit file "/etc/apache2/sites-enabled/000-default", and add the following at its end, just before </VirtualHost>, for *Apache* to delegate HTTP requests to the sample *Tomcat* web applications to *Tomcat* (removing the same lines we inserted earlier so there would be no duplicate lines):

```
   ProxyPass  /examples/servlets  ajp://localhost:8009/examples/servlets
   ProxyPassReverse /examples/servlets jp://localhost:8009/examples/servlets

   ProxyPass  /examples/jsp  ajp://localhost:8009/examples/jsp
   ProxyPassReverse  /examples/jsp  ajp://localhost:8009/examples/jsp

   ProxyPass  /docs  ajp://localhost:8009/docs
   ProxyPassReverse  /docs  ajp://localhost:8009/docs
```

©Copyright 2009 Prof. Lixin Tao

```
ProxyPass  /survey  ajp://localhost:8009/survey
ProxyPassReverse  /survey  ajp://localhost:8009/survey

ProxyPass  /surveyJsp  ajp://localhost:8009/surveyJsp
ProxyPassReverse  /surveyJsp  ajp://localhost:8009/surveyJsp

ProxyPass  /surveyCss  ajp://localhost:8009/surveyCss
ProxyPassReverse  /surveyCss  ajp://localhost:8009/survetCss

ProxyPass  /surveyJsf  ajp://localhost:8009/surveyJsf
ProxyPassReverse  /surveyJsf  ajp://localhost:8009/surveyJsf

ProxyPass  /demo  ajp://localhost:8009/demo
ProxyPassReverse  /demo  ajp://localhost:8009/demo

ProxyPass  /tripler  ajp://localhost:8009/tripler
ProxyPassReverse  /tripler  ajp://localhost:8009/tripler

ProxyPass  /testCookie  ajp://localhost:8009/testCookie
ProxyPassReverse  /testCookie  ajp://localhost:8009/testCookie

ProxyPass  /bookstoreServlet  ajp://localhost:8009/bookstoreServlet
ProxyPassReverse  /bookstoreServlet  ajp://localhost:8009/bookstoreServlet

ProxyPass  /bookstoreJsp  ajp://localhost:8009/bookstoreJsp
ProxyPassReverse  /bookstoreJsp  ajp://localhost:8009/bookstoreJsp

ProxyPass  /axis  ajp://localhost:8009/axis
ProxyPassReverse  /axis  ajp://localhost:8009/axis

ProxyPass  /bareJsp  ajp://localhost:8009/bareJsp
ProxyPassReverse  /bareJsp  ajp://localhost:8009/bareJsp

ProxyPass  /guessNumber  ajp://localhost:8009/guessNumber
ProxyPassReverse  /guessNumber  ajp://localhost:8009/guessNumber

ProxyPass  /html  ajp://localhost:8009/html
ProxyPassReverse  /html  ajp://localhost:8009/html
```

Now you can use *Apache* to visit the sample *Tomcat* web applications in the next subsection without using port number specification. We can bypass *Apache* and visit these applications directly on *Tomcat* by adding the port number 8080 specification, as in http://localhost:8080/demo".

#### 4.15.2.3 Testing Sample *Tomcat* Web Applications

Use *Firefox* to visit the following URLs to test the web applications that we deployed in the last step:

- http://localhost/examples/servlets
- http://localhost/examples/jsp
- http://localhost/survey
- http://localhost/surveyJsp
- http://localhost/surveyCss

- http://localhost/surveyJsf
- http://localhost/demo
- http://localhost/html
- http://localhost/tripler
- http://localhost/testCookie
- http://localhost/bookstoreServlet
- http://localhost/bookstoreJsp
- http://localhost/axis
- http://localhost/bareJsp
- http://localhost/guessNumber