Experiment 5

Experiment Title:

Implementation of Para-Virtualization using VMware’s workstation/ Oracle’s Virtual Box and Guest OS.

Aim:

Implementation of Virtual Box for virtualization of any OS.

Theory:

Virtual Box is a cross-platform virtualization application. What does that mean? For one thing, it installs on your existing Intel or AMD-based computers whether they are running Windows, Mac, Linux or Solaris operating systems. Secondly it extends the capabilities of your existing computer so that it can run multiple operating systems at the same time. You can install and run as many virtual machines as you like the inly practical limits are disk space and memory. Virtual Box is deceptively simple yet also very powerful. It can run everywhere from small embedded systems or desktop class machines all the way up to datacentre deployments and even Cloud environments. The techniques and features that Virtual Box provides are useful for several scenarios.

Running multiple operating systems simultaneously:

Virtual Box allows you to run more than one operating system at a time. This way you can run software written for one operating system on another without having to reboot to use it. Since you can configure what kinds of “virtual” operating system such as DOS or OS/2 even if your real computer’s hardware is no longer supported by that operating system.

Easier software installation:

Software vendors can use virtual machines to ship entire software configurations. With Virtual Box such a complex setup can be packed into a virtual machine. Installation and running a mail server become as easy as importing such an appliance into Virtual Box.

Testing and disaster recovery:

Once installed a virtual machine and its virtual hard disks can be considered a “container” that can be arbitrarily frozen woken up copied backed up and transported between hosts.

Infrastructure consolidation:

Virtualization can significantly reduce hardware and electricity costs. Most of the time computers today only use a fraction of their potential power and run with low average system loads. A lot of hardware resources as well as electricity if thereby wasted. So instead of running many such physical computers that are only partially used one can pack many virtual machines onto a few powerful hosts and balance the loads between them

Some terminologies used:

When dealing with virtualization it helps to acquaint oneself with a bit of crucial terminology especially the following terms:

Host Operating System:

This is the operating system of the physical computer on which Virtual Box was installed. There are versions of Virtual Box for Windows, Mac, Linus and Solaris hosts.

Guest Operating System:

This is the operating system that is running inside the virtual machine. Theoretically Virtual Box can run any x86 operating system but no achieve near-native performance of the guest code on your machine we had to go through a lot of optimizations that are specific to certain operating systems. So while your favourite operating system may run as a guest we officially support and optimize for a select few.

Virtual Machine:

This is the special environment that Virtual Box creates for your guest operating system while it is running. In other words you run your guest operating system “in” a VM. Normally a VM will be shown as a window on your computers but depending on which of the various frontends of Virtual Box you use it can be displayed in full screen mode or remotely on another computer. In a more abstract way internally Virtual Box thinks of a VM as a set of parameters that determine its behaviours. They include hardware settings as well as state information. These settings are mirrored in the Virtual Box Manager window as well as the VBoxManage command line program.

Guest Additions:

This refers to special software packages which are shipped with Virtual Box but designed to be installed inside a VM to improve performance of the guest OS and to add extra features.

Starting Virtual Box:

After installation you can start Virtual Box as follows:

1. On a windows host in the standard “Program” menu click on the item in the Virtual Box group. On Vista or Windows 7 you can also type “VirtualBox” in the search box of the Start menu.
2. On a Mac host in the Finder double-click on the “VirtualBox” item in the applications folder.
3. On a Linux or Solaris host depending on your desktop environment a “VirtualBox” item may have been placed in either the “System” or “System Tools” group of your application menu. Alternatively you can type “VirtualBox” in a terminal.

Creating your first Virtual Machine:

Click on “New” button at the top of the VirtualBox Manager window. A wizard will pop up to guide you through setting up a new virtual machine.

One the following pages the wizard will ask you for the bare minimum of information that is needed to create a VM in particular:

The VM name will later be shown in the VM list of the VirtualBox Manager window and it will be used for the VM’s files on disk. Even though any name could be used keep in mind that once you have created a few VMs you will appreciate if you have given your VMs rather informative names “Mt VM” would thus be less useful than “Windows XP SP2 with OpenOffice”.

For “Operating System Type” select the operating system that you want to install later. The supported operating systems are grouped of you want to install something very unusual that is not listed select “Other”. Depending on your selection Virtual Box will enable or disable certain VM settings that your guest operating system may require. This is particularly important for 64-bit guests. It is therefore recommended to always set it to the correct value.

On the next page select the memory that Virtual Box should allocate every time the virtual machine is started. The amount of memory given here will be taken away from your host machine and presented to the guest operating system which will report this size as the computer’s installed RAM

A windows XP guest will require at least a few hundred MB RAM to run properly and Windows Vista will even refuse to install with les than 512 MB. Of course if you want to run graphics-intensive applications in your VM you may require even more RAM.

So as a rule of thumb if you have 1 GB of RAM or more in your host computer it is usually safe to allocate 512 MB to reach VM. But in any case make sure you always have at least 256 to 512 MB of RAM left on your host operating system. Otherwise you may cause you host OS to excessively swap out memory to your hard disk effectively bringing your host system to a standstill. As with the other settings you can change this setting later after you have created the VM.

1. Next you must specify a virtual hard disk for your VM. There are many and potentially complicated ways in which Virtual Box can provide hard disk space to VM but the most common way is to use a large image file on your “real” hard disk whose contents Virtual Box presents to your VM as if it were a complete hard disk. This file represents an entire hard disk then so you can even copy it to another host and use it with another Virtual Box installation.

Here you have the following options:

To create a new empty virtual hard disk press the “New” button. You can pick an existing disk image file. The drop-down list presented in the window contains all disk images which are currently remembered by Virtual Box probably because they are currently attached to a virtual machine. Alternatively you can click on the small folder button next to the drop-down list to bring up a standard file dialog which allows you to pick any disk image file on your host disk.

Most probably if you are using Virtual Box for the first time you will want to create a new disk image. Hence press the “New” button. This brings up another window the “Create New Virtual Disk Wizard” which helps you create a new disk image file in the new virtual machine’s folder. Virtual Box support two types of image files:

1. A dynamically allocated file will only grow in size when the guest actually stores data on its virtual hard disk. It will therefore initially be small on the host hard drive and only later grow to the size specified as it is filled with data.
2. A fixed-size file will immediately occupy the file specified even if only a fraction of the virtual hard disk space is actually in use. While occupying much more space a fixed-size file incurs less overhead and is therefore slightly faster than a dynamically allocated file.

After having selected or created your image file again press “Next” to go to the next page.

After clicking on “Finish” your new virtual machine will be created. You will then see it in the list on the left side of the manager window with the name you entered initially.

Running your virtual machine:

To start a virtual machine you have several options:

1. Double-click on its entry in the list within the Manager window or select its entry in the list in the Manager window it and press the “Start” button at the top
2. For virtual machines created with Virtual Box 4.0 or later navigate to the “Virtual Box VMs” folder in your system user’s home directory find the subdirectory of the machine you want to start and double-click on the machine settings file. This opens up a new window and the virtual machine which you selected will boot up. Everything which would normally be seen on the virtual system’s monitor is shown in the window. In general you can use the virtual machine much like you would use a real computer. There are couple of points worth mentioning.

Saving the state of the machine:

When you click on the “Close” button of your virtual machine window Virtual Box asks you whether you want to “save” or “power off” the VM.

Save the machine state:

With this option Virtual Box “freezes” the virtual machine by completely saving its state to your local disk. When you start the VM again later you will find that the VM continues exactly where it was left off. All your programs will still be open and your computer resumes operation. Saving the state of a virtual machine is thus in some ways similar to suspending a laptop computer.

Send the shutdown signal:

This will send an ACPI shutdown signal to the virtual machine which has the same effect as if you had pressed the power button on a real computer. So long as the VM is running a fairly modern operating system this should trigger a proper shutdown mechanism from within the VM

Power off the machine:

With this option Virtual Box also stops running the virtual machine but without saving its state. As an exception if your virtual machine has any snapshots you can use this option to quickly restore the current snapshot of the virtual machine. In that case powering off the machine will not disrupt its state but any changes made since that snapshot was taken will be lost. The “Discord” button in the Virtual Box Manager window discards a virtual machine’s saved state. This has the same effect as powering it off and the same warning apply.

Importing and exporting virtual machines:

Virtual Box can import and export virtual machines in the industry-standard Open Virtualization Format (OVF). OVF is a cross-platform standard supported by many virtualization products which allows for creating ready-made virtual machines that can then be imported into a virtualize such as Virtual Box. Virtual Box makes OVF import and export easy to access and supports it from the Manager window as well as its command-line interface. This allows for packaging so-called virtual appliances: disk images together with configuration settings that can be distributed easily. This way one can offer complete ready-to-use software packages that need no configuration or installation except for importing into Virtual Box.

Appliances in OVF format can appear in two variants:

1. They can come in several files as one or several disk image typically in the widely used VMDK format and a textual description file in an XML dialect with an .ovf extension. These files must then reside in the same directory for Virtual Box to be able to import them.
2. Alternatively the above files can be packed together into a single archive file typically with an .ova extension.

Select File → Export appliance. A different dialog window shows up that allows you to combine several virtual machines into an OVF appliance. Then select the target location where the target files should be stored and the conversation process begins. This can again take a while.

Conclusion:

Thus we have studies the use of Multiple Operating Systems by virtualization using Virtual Box.