

# Virtual Machines: What They Are and What They Can Do

“A safe and easy way to learn and use a new OS.”

*by Pete Choppin*

Most computer users think of virtual machines in the "virtual reality" sense. Actually, that perception is only slightly off track. Virtual machines have little in common with virtual reality, but what they share with VR is a simulation function. Virtual reality is a simulation of reality, while virtual machines are a simulation of a particular machine, in most cases the machine you will be simulating/emulating is a standard PC. However, they can be used to simulate a host of other computer hardware based on a wide array of target processor architectures.

"Why would I want to simulate a PC when I already have a PC?" That is an excellent question.

## Common Uses of VMs

There are many reasons to use a VM. Here are just a few:

- Test out new operating systems, such as Linux, which we keep hearing so much about, or even run Windows on a Mac or Linux machine.
- Test or run applications designed for other hardware. For example, running applications written for a phone or embedded system.
- Test out unknown applications that may cause damage to your operating system's stability. Programmers will often use this technique to discover bugs in an application that could bring the whole system down, without bringing the whole system down.
- Run test applications before putting them into production. A virtual machine is ideal for simulating the operating systems or servers that run in a production environment. It is far easier and much less costly to test your application in a virtual environment rather than risk downtime in your production environment. Restoring the virtual machine after testing is nothing more than reloading the original files, which is far less of a problem than restoring a server or desktop computers.

## Drawbacks to Using Virtual Machines

Generally, emulating an entire machine in software is almost always going to be slower than actual hardware. (Some newer processors have specialized hardware designed to accelerate virtualization. Not all VMs take advantage of them, but when they do you can achieve near-native speeds.)

Interaction with a virtual machine and the host OS is difficult if at all possible. Copying and pasting from a Windows document to a Linux VM is almost always unsupported.

Accessing files or playing music from a folder on the host in the VM can be tricky. Most VMs emulate a network adapter, so you can share files between the host and guest as if it were another machine on the network.

## Which Virtual Machine Software Should I Use?

There is a wide array of virtual machine software and other related applications—so many, in fact, that some may consider it a dizzying array. For our purposes, we will focus on a few of the more popular freely available options, listed in no particular order.

### VMPlayer:

Free to download at no cost for personal use  
Hosts: Windows and Linux compatible

VMPlayer will allow you to launch preconfigured "[virtual appliances](#)" as well as create your own.

### VMWorkstation:

Commercial software available from [VMWare](#)  
Hosts: Windows and Linux compatible

This is a fully featured virtual machine application for creating and running virtual machines. VMWorkstation handles multiple VMs at a time and is equipped with special features such as snapshots, USB support and the ability to handle up to two processors in a guest. VMWorkstation also supports a feature called Teams—gangs of virtual machines that can be powered on and off together, networked together automatically, and interact in predefined ways. An application for this is a multitier client/server/database setup.

Be sure to install the VMware tools on your guest. They provide a set of special drivers and utilities that the program can install automatically on most guest operating systems. When these tools are installed, they optimize multimedia performance and seamlessly allow VMware to detect when your mouse is inside the virtual machine window and act appropriately.

### VirtualBox:

Free (100%) to download  
Hosts: Windows, Linux, Mac, Solaris compatible

VirtualBox—another fully functional virtual machine application. Made by Sun Microsystems (now Oracle), VirtualBox supports snapshot images so that you can go back to any point and reload your VM instantly. A virtual disk manager lets you easily manage all your images.

VirtualBox is quite stable. I have been able to run VMs on VirtualBox where VMWare had problems.

## Let's Set Up a Virtual Machine

Now that we have looked at a few virtual machine applications, it is time to create a virtual machine. For this demonstration, I will be installing a virtual Windows 7. Please note that even in a virtual environment, Microsoft requires that you purchase a valid license of its operating system.

This install will be running on Fedora Linux, and we will be using VirtualBox for this example.

VirtualBox is fairly easy to install in Linux, and virtual machine creation is very straightforward.

For this installation, you will need a licensed copy of Windows 7, the VirtualBox software installed on Linux, and ample hard drive space available (four to 10 gigabytes should suffice).

If you prefer to run VirtualBox on a Windows host, the virtual machine creation is almost identical. In fact, most of the setup I am demonstrating is the same, regardless of what operating system you choose for the host or the guest.

The first step is to start VirtualBox. A welcome screen appears that begins the setup wizard to create your virtual machine.

The screens in the wizard are fairly self-explanatory with on-screen instructions. Essentially, it walks you through the entire VM creation process.

A couple of concepts for VM creation are important. There is something called a virtual hard disk that the software sets aside in a file. It emulates a physical hard drive on which the guest operating system is installed. Other files are also created that have settings for the virtual machine, such as a network interface, video adapter, sound, USB, etc.—all components that make up your virtual machine.

After you complete the wizard, you are brought to the VirtualBox screen where you can manage all your virtual machines. All of the details about each VM are accessible and can be modified in VirtualBox.

Once the VM is created, you can select the VM you want to start and then simply click the Start button to launch it.

The VM will launch as though you are booting it up on an actual hard drive. Once it boots, your VM will function just like a computer running on its own hardware.

One of the nicest features of VirtualBox (and VMWare for that matter) is the ability to create snapshots of the OS. When you have a snapshot, you can go back to that state at any time, reversing anything you installed or anything that is no longer working after the snapshot was taken. It is a good idea to take your first snapshot soon after the OS is first installed so that you have a known good restore point to go back to in case you break something.

Image 1. VirtualBox snapshot dialog box.

## Image 2. VirtualBox snapshot list.

In Image 1, I create the first snapshot, and Image 2 shows several snapshots that I have taken. I can launch any of these that I need, giving me the ability to go back to any point at which I have taken a snapshot.

This makes installing and testing software very easy and safe. I can go back to any point I want and undo or redo whatever I need. And it is all in a virtual machine, not in production or being affected in any way by what I am doing. An additional advantage is that no additional hardware needs to be used for testing. I can create many virtual machines, using different operating systems, all on the same hardware.

Virtual machines are the ideal solution for software testing, or even just for learn new operating systems such as Linux. They are safe, easy to use and restore, inexpensive and have many uses. In fact, once you start using virtual machines, you will wonder why you ever considered trying out software on a real computer.