VMware® Infrastructure 3

Advanced Technical Design Guide

~and~

Advanced Operations Guide

Two books in one!

Ron Oglesby
Scott Herold
Mike Laverick

Chapter 5: Create and Modify VMs

In this chapter I will cover how to create a VM for the first time. Once we have created our first VM, Chapter 6 will address methods of duplicating the VM to save time.

I will define in more detail the virtual hardware available within a VM; examine what it is like to setup Windows and Linux within a VM, and how to install VMware Tools to your guest operating system. I will also address how to add additional "hardware" to the virtual machine – and show how it is possible to enable direct access to a SAN or iSCSI system within a VM.

I will cover how to use the new "Snapshot" facility introduced into ESX 3.x. The snapshot facility allows you to undo operations within a VM and is used in the process of backing up a VM while the VM is powered on. I cover how to un-register, remove, and delete a VM. Lastly, I will give you a quick jumpstart on how to create a VM from an existing physical server using VMware Converter.

Before you jump right in and start installing a VM you should really check out the list of supported guest operating systems, especially if your guest is a not Microsoft Windows. With operating systems like Solaris, Novell, Linux, and FreeBSD there can often be quite a surprising gap between what is officially supported by VMware and the current distribution.

http://www.vmware.com/pdf/GuestOS_guide.pdf

What defines a VM?

As you might know already a VM is firstly a collection of files. In more detail the actual files are

- .vmx – Configuration file in text format
- .nvram – The VMs virtual BIOS file

- .vmdk – The VMs metadata/descriptor virtual disk file
- .flat-vmdk – The VMs data virtual disk file (OS/Apps, Data)
- .vswp – The VMs swap file
- .delta – The Snapshot file
- .vmsn – The Snapshot Memory File
- .vmsd – The Snapshot Manager File
- .log – Log file
- .vmxf – Some kind of Internal Metadata file
- .rdm – RAW Device Mapping file with Virtual Compatibility
- .rdmp – RAW Device Mapping file with Physical Compatibility

When you create a VM what you're actually doing is creating a text file with the VMX extension. The VMX file will hold the definition of the VM properties which include:

- VM's name
- Storage location
- Guest Operating System type
- Number of virtual CPUs (vCPU)
- Number of virtual NICs and the port groups they connect to
- Type of virtual SCSI adapter used
- Size and location of virtual disk

The VM itself presents the appearance of real hardware even though we know it is actually software – a virtual machine. When the guest operating system makes a hardware request it believes the VM is actually a physical machine with a physical motherboard. An ESX VM actually uses an Intel 440BX-based virtual motherboard with an NS338 Chip. VMware selected this motherboard because it has good compatibility and reliability with all the guest operating systems supported by ESX – so it can even cope with something very old like Windows NT 4.x. This selection of the motherboard then defines the virtual

hardware that can be used within it. So what are the options allowed by this virtual motherboard? Here's a quick list of what is supported:

- 1-2 virtual floppy drives
- 1-2 virtual CD/DVD drives
- PS/2 interfaces for keyboard and mouse
- 6 PCI slots with the 6th used by the virtual video adapter – leaving 5 left over for you to configure.
- 1-4 vCPUs
- Up to 16GB of RAM
- 1 Parallel Port
- 1-2 Serial Ports

You will notice that there is no support for sound or USB. For the most part this is not a problem; if you use terminal services to connect to Windows running inside a VM you can have sound redirected to the client device. The lack of USB support (which is available in VMware Server and Workstation VMs) could cause a problem if you are running software that requires a "dongle" for licensing purposes. The most common solution is to purchase an IP enabled USB hub and redirect the USB calls to the network.

Parallel and serial devices are not fully virtualized, and their functionality is provided by the Service Console, not the VMkernel. There is a huge drawback to configuring parallel and serial devices in this way. Firstly, you will be limited by the number of physical parallel and serial ports at the back of the ESX host. You will be trouble if you have more VMs that require this kind of hardware than you have physical ports available. Secondly, if you did configure this you would be unable to VMotion that VM. VMotion, if you remember, is the process of moving a VM (while powered on) from one ESX host to another. VMware ESX server is very clever but not clever enough to unplug a dongle from the back of one ESX host and plug it in the back of another during VMotion. So again, many people purchase an IP enabled parallel or serial hub and redirect the hardware calls through the network instead.

Virtual CD and Floppy

In the world of ESX we very rarely use physical CD/DVDs and floppies. Generally, we take CD/DVDs and floppy drives and convert them into ISO or FLP files. There are a number of tools that will help you do this. After a CD or floppy disk has been "ripped" to an ISO file we then upload this to NAS, SAN, or iSCSI storage, depending on our preference.

To create an ISO file you can use tools such as your CD burning software, WinISO, MagicISO, or WinImage. There are tools in Linux such as the 'dd' and the 'mkisofs' command which will create ISO's images for you, but you should be aware that the 'dd' command does not verify that the ISO is a perfect image of your physical CD.

To copy these files to a storage address by ESX, you can use a free application called WinScp. This allows you to connect to the ESX host and then, using an explorer-like interface, drag-and-drop files to the relevant storage location. Unfortunately, for this tool to work, you would need to lower the security for the SSH daemon in ESX. This is because SSH access for root is blocked by default. However, the free WinSCP tool does not allow you to login as a lower-level user and then elevate your privileges to root. To lower security to allow root access, open a SSH session to your ESX host and then:

1. Type the command:

   ```
   nano –w /etc/ssh/sshd_config
   ```

2. Modify

   ```
   PermitRootLogin no
   ```

   to be:

   ```
   #PermitRootLogin no
   ```

3. Save the configuration file and exit nano
4. Then restart the ssh service with

   ```
   service sshd restart
   ```

Note:

In recent months I have switched away from WinSCP to Veeam SCP for ESX. At the time of writing this chapter the tool was free although you do have register an email address to download it. I've found Veeam significantly faster than WinSCP.

Keyboard and Mouse Interface

Of course, VMware has yet to produce the VMware Keyboard and Mouse like Microsoft. After all, these PS-2 connections don't physically exist; therefore you cannot plug in a physical keyboard and mouse. To interact with a VM you open a "Remote Console" session. This is similar in functionality to an ILO or RAC card or IP KVM on a physical machine. It allows you to watch a VM boot up just like a physical machine. It also allows you to send keyboard and mouse movements from your management PC to the VM.

Virtual Video Adapter

During the installation of your guest operating system the graphics will be quite poor and you might also experience poor mouse performance. During the installation a standard VGA driver is used. After the guest operating system has completed its installation we can install VMware Tools. Amongst many other things, VMware Tools adds a VMware Virtual Video Adapter Driver and VMware Virtual Mouse Driver. These significantly improve graphics and mouse operation, especially in operating systems like Windows that cannot be run without a GUI front-end.

Creating a VM

In this section I will guide you through the creation of your first VM, stopping along the way to point out some handy tips and tricks and explain some of the less obvious options in the Wizard.

GOTCHA:

One of the most common "mistakes" made by people new to virtualization is that they create VMs with the same amount of virtual hardware as they do for a physical hardware. Every VM this person creates is defined with 4 vCPUs, 4GB RAM, and 72GB virtual disks – even if the VM only uses 10% of these resources. A better practice is to define the VM with the minimums that you feel your applications or services can run with. Resources can always be increased afterwards if you set them too low. The golden rule is that it is always easier to give away resources on a need-to-use them basis than it is to take back needlessly allocated resources. A good analogy is for this is permissions. We only give the users the permissions they need as it is always easier to grant more privileges than it is to take privileges away. In this respect, this issue is also about setting reasonable expectation. Allocating too many resources to the VMs your operators manage will then set the expectation that all VMs should be configured in this way.

I will start creating my VM in the "Host and Clusters" view by selecting one of my ESX hosts.

Creating a VM

1. Select your ESX host.
2. Right-click and choose New Virtual Machine.
3. Choose Custom (so you see all options).
4. Type in a friendly name for your VM.

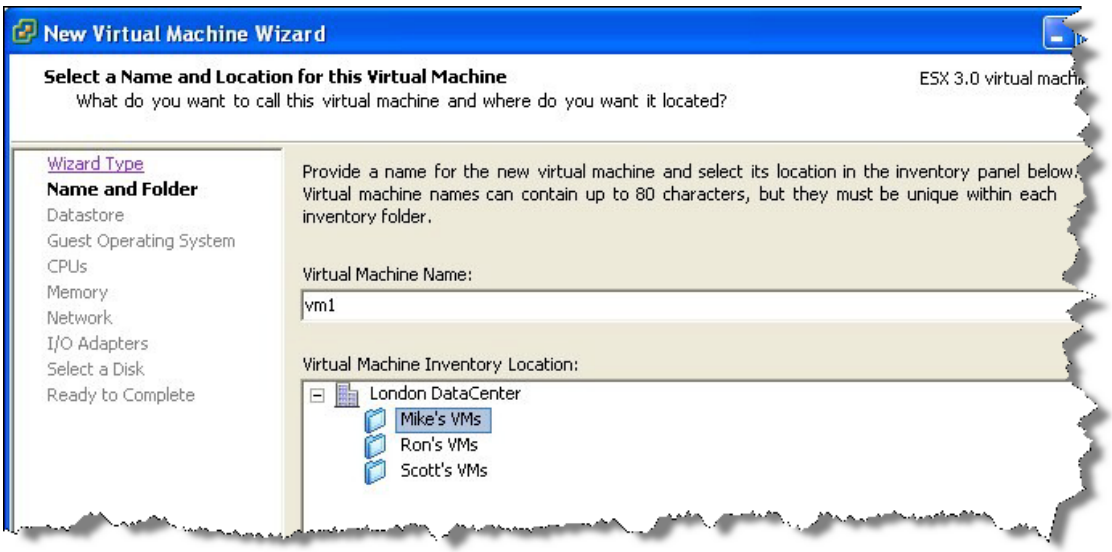
   Note: Use the Right Characters

   I recommend avoiding spaces and special characters and restricting yourself to using only alphanumeric characters. I advise using just lower-case as well. If you are ever at the command-line you may have to type the name of the VM. This is much harder if you have used a mix of cases, special characters, and spaces.

   In this dialog box you should also see the datacenter name within which your ESX host is located. If you have created folders for virtual machines in the "Virtual Machines and Templates View"

you will be able put the VM into the relevant location. Figure 5.1 shows me selecting the correct folder to hold my VM.

Figure 5.1



5. Next we Choose a Datastore to for the VM's files.

   Note:

   This dialog box tells you two very useful things. Firstly, it tells you how much free space you have on a datastore. Virtual disks can be quite large files so you will want to select a datastore that has enough free space for your VM. Secondly, the "Access" columns should tell you if the datastore is available to more than one ESX host – or on a shared storage. Shared storage is a requirement for VMotion, VMware DRS, and VMware HA.

6. From the pull-down list, Choose the Guest Operating System that will be run inside the VM.

   Note: Why select the right operating system type?

   This is important for two reasons. Firstly, it will assist the system when you install VMware Tools. VMware Tools ship as a Microsoft Installer package for Windows guests and as a Redhat Package Management (RDM) file and as a.gz zip file for Linux guests. By selecting the correct operating system in the list you will find the right version of VMware Tools is installed to the guest OS. Secondly, selections of the guest OS will sometimes dictate the re-

maining defaults in the wizard. For example, if you select Windows 2000 from the list, the default for the virtual SCSI adapter will be a Buslogic device. If you select Windows 2003 from the list this default changes to LSIlogic.

7. Configure the number of vCPUs in the VM for the guest operating system.

   Note:

   I would recommend starting with 1 vCPU and then adding more if you think later on that it might assist. Microsoft does not officially support "downgrading" from many CPUs to 1 CPU. There are of forcing a downgrade of CPUs. These methods are not officially supported although they frequently do work.

   Adding an extra vCPU does not necessarily improve performance, especially if your applications or services are not "multi-threaded." In order to leverage the real benefit of multiple vCPUs, the physical hosts may need physical sockets or cores. This is because of the way the VMkernel schedules processes that are to be executed. I will elaborate more on this subject in chapter 7 when we cover Resource Monitoring and Management.

8. Next Configure the VM's memory size.

9. The amount set here will act as a limit or maximum to be allocated to the VM. Even if you have free memory available, the VM will never exceed this amount.

10. Next choose which network connections will be used in the VM.

    Note:

    It is possible to configure a VM without a NIC at all. The usefulness of such a VM would be pretty limited, but it is an option that is available.

11. Next we select the Storage I/O Adapter type.

    Note:

    There are two types of virtual SCSI adapter in the VM – BusLogic and LSIlogic. Windows NT and Windows 2000 default to the BusLogic, whereas Windows XP, Windows 2003, and Windows

Vista default to LSIlogic. In contrast most Linux distributions default to the LSILogic Driver.

12. Next, in the Select a disk dialog, we choose Create a new virtual disk.

13. The option to "Use an existing virtual disk" could be selected if you had copied a virtual disk from another VMware product like VMware Workstation or Server (this process is covered in Chapter 12: ESX on the Command-Line). The "Mapped SAN LUN" option is used to give a VM direct access to SAN or iSCSI LUN. Later in this chapter I will cover this option in the "Adding Devices to a VM" section.

14. Next Specify Disk Capacity and Location.

    Note:

    When a virtual disk is created it takes up ALL the space you allocate here. ESX virtual disks do not "grow" as more data is created inside them. The term that is frequently used for this format is "monolithic" virtual disks. This format offers the best performance – as a flat file the virtual disk will be created in contiguous blocks within the VMFS volume.

    How big should a virtual disk be? Well, it depends on what you are putting into it. One gotcha is giving a VM a large amount of memory, but a very small virtual disk for the operating system. You might find you lack space for the swap space for your given guest operating system. Now that the amount of memory is a configurable value this needs some consideration. What free space would there be for a swap file or partition if you started with 250MB of RAM and then later changed to 2GB RAM? Virtual disks can be made larger, and there are methods for making them smaller. I cover these in Chapter 12: *ESX on the Command-Line.*

15. Finally, the option to "Specify a datastore" allows you to store the VM's virtual disks at an alternative location. Perhaps you have two virtual disks – one for the boot disk and the other for data. Many people like to put their boot disks and data disks on different LUNs which have different RAID or backup levels.

16. Specify any Advanced Options, choose SCSI 0:0.

Note:

In SCSI systems, adapter 0 and id 0 are used to indicate the location of the boot disk (SCSI 0:0). A VM conforms to all the SCSI conventions with the adapter using SCSI ID7. If you remember the VM presents to the guest operating system the appearance of real hardware. You can have up to 4 virtual SCSI adapters if you wish, with up to 15 virtual disks attached to each adapter. Although the range is from 0:0 to 0:15 (which is actually 16 SCSI ids), id 0:7 is used by the controller itself.

I will cover the mode options of "independent," "persistent," and "non-persistent" when I delve into the "snapshot" feature later in this chapter.

17. Lastly, check your selections in the summary page – and click Finish.

First Power on and Installing the Guest Operating System

When you first power on a VM its boot order is:

1. Floppy
2. CD
3. Hard Disk
4. PXE

The second time you power on a VM the boot order is changed to:

1. Floppy
2. Hard Disk
3. CD
4. PXE

As you can see, on the first boot up the assumption is that you will probably be booting to a CD-ROM ISO to install the guest operating system. The second time you boot it is assumed that the VM is going to boot from the hard disk.

This stops the annoyance created by some vendors' operating systems where their CDs do allow the operator to skip booting from the CD. It also means that if you want an easy life you should really connect an ISO to the VM *before* the first power on to avoid having to use the VM's BIOS [F2] or [ESC] key-strokes to change the boot order. If you fail to connect a CD at the first boot you will find that the VM boots to PXE and searches for a DHCP server. If, subsequently, the CD is connected and then VM rebooted – because the boot order has been changed the VM would still not boot to the CD!

To attach an ISO to the VM you have 3 choices:

- You can use the ESX server's physical CD
- the CD-ROM drive or ISO on your management PC
- or an ISO on a centralized datastore.

The last option is the best in terms of what's best for performances, flexibility, and low administration cost. As discussed previously in the storage chapter, the ISO could be copied to an SAN, iSCSI, or NAS storage.

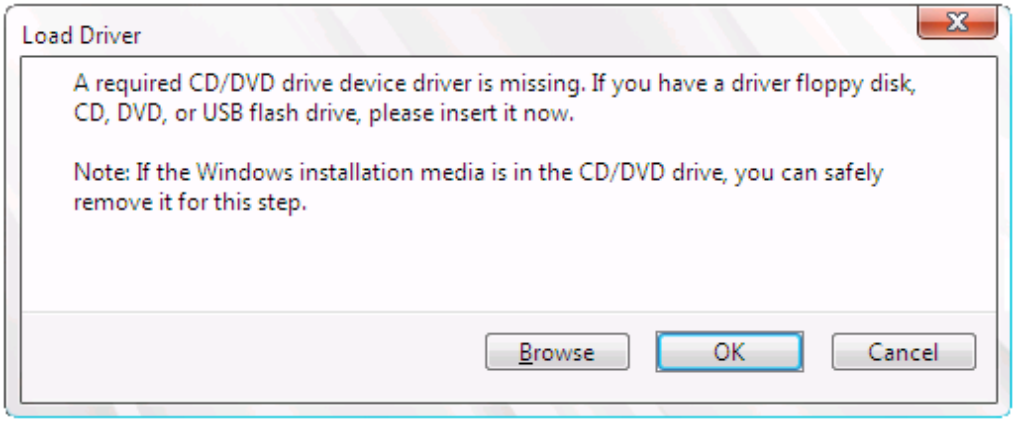
1. Right-click your new VM.
2. Choose **Edit Settings.**
3. In the list of devices choose **CD/DVD Drive 1.**
4. Choose the option **Datastore ISO file.**
5. Select a datastore where you have copied the ISOs of your guest operating system – and select the ISO.
6. Before leaving the "**Virtual Machine Properties**" box, ensure you enable the option "**Connect at Power On.**"
7. To open a Window on the VM, **right-click your VM,** and choose **Open Console.**
8. You can power on the VM using **VM, Power On** in the console window**.**

Installing Windows Vista

Installing Windows Vista is just like installing any other flavor Windows. The new version of Windows needs a terrific amount of memory and disk space to perform well. I gave my first installation 1GB of memory and 16GB virtual disk. After a plain vanilla installation I had nearly 512MB RAM when running idle and used 7.15GB of disk space.

There are, however, issues currently outstanding with running Vista on ESX 3.0.1 and older. Although the BIOS of the VM is able to boot the ISO of Vista, during the Windows installation Vista fails to find the CD/DVD which VMware emulate. The problem has been since been resolved in ESX 3.5. Figure 5.2 shows the error message in question:

Figure 5.2



It appears that there is no driver on the Vista DVD. I came across this issue in my first installation of Vista and had to use the forums to find a work-around as currently there is not a KB article from VMware. There are 3 main ways of fixing this issue. Helpfully, some people have already put the driver into a flp file for us. It is located here:

http://sti.epfl.ch/intranet/informatique/virtualisation/drivers-vista-rtm-esx.flp.zip

http://www.rtfm-ed.co.uk/downloads/winvistacddrivers.flp

If you want to read the original forum post about this issue it is located here:

http://www.vmware.com/community/thread.jspa?threadID=62141

To use this new driver, wait until you receive the Load Driver prompt, then:

1. **Press Ctrl+Alt** to release your keyboard and mouse.
2. In the VM console window, Choose **VM** and **Edit Settings.**
3. Select **Floppy Drive 1.**
4. Choose **Use existing floppy image in the datastore**, and click the **Browse** button.
5. **Select the FLP image downloaded** from above locations listed.
6. Select **Connected.**
7. Click back into the **VM console window**, and click **OK.**

   Note:

   The Windows Vista installer should read from the floppy disk and find CD-ROM Drive (A:cdrom.inf).
8. Click **Next,** and continue with the installation.

VMware Tools

After the operating system is installed, we generally install VMware Tools. This is a software package that is installed to the guest operating system. It contains three components.

- Drivers
- Service or daemon
- Configuration Applet or script

Drivers

During the installation of VMware Tools, the installer copies across 6 drivers for the following devices: VMware SVGA II, VMware Pointing Device, VMware SCSI Driver (replaces Microsoft BusLogic Driver if used in Windows NT and 2000), AMD Enhanced NIC Driver (vmxnet.sys replaces the

pcntpci5.sys), a file system synchronization driver (used during VCB backups), and a memory control driver (vmmemctl). These devices and the drivers that accompany them significantly improve performances, especially the vmxnet.sys network card driver. Therefore VMware Tools is highly recommended. The memory control driver is used to control memory usage when physical RAM is scarce. I will discuss in more detail this driver in the performance chapter.

Service or Daemon

The "heartbeat" service or daemon is installed as part of VMware Tools. This service is used to alert the administrator that the guest operating system inside the VM has malfunctioned. Under normal operation a VM should have a small green icon next to it in the Inventory. If a VM "hangs," blue-screens (BSOD), or in Linux experiences a kernel panic, you should see this icon change to a red exclamation mark next to the VM. The guest operating system error stops the VMware Tools heartbeat service which then triggers an alert or an alarm.

You may get benign alerts occurring when you first power on a VM because the guest operating system is still loading, and the heartbeat service has yet to start.

Configuration Applet or Script

If you install VMware Tools to Windows you should find you have an icon in the taskbar tray near the clock. If you install VMware Tools to Linux without a GUI then you have a script that you can run instead. The applet or script is used to configure VMware Tools after the installation process has completed. VMware Tools have many configuration options available including:

1. Enabling Time Synchronization
2. Scripts triggered by power events
3. Modify Connected Devices

Installing and Configuring VMware Tools for Windows

1. Logon to your Windows VM.
2. In the **Console** window, choose **VM** in the menu.

3. Select **Install VMware Tools.**
4. Click **OK** to the dialog box.

   Note:

   In the background the system connects to ISO called window.iso held in /usr/lib/vmware/isoimages/windows. Windows should autorun this CD and execute the "VMware Tools.msi" file.

5. Choose **Typical.**

   Note: Typical Vs Complete Vs Custom

   A typical installation only configures features that are required with VMware ESX. A complete installation configures features for all VMware platforms – ESX, Server, and Workstation. A custom installation allows you to select which components outlined above you require. Interestingly there is a "hidden" driver option in the custom installation called "Descheduled Time Accounting." This driver only has experimental support at the time of writing. It is used to improve the quality of time synchronization in a VM.

Installing and Configuring VMware Tools for Linux

In Linux there are two VMware Tools packages. The first is in a Redhat Package Management (RPM) format. The second is in zipped format of tar.gz. After extracting the tar.gz file to a temporary location a script is used to install and configure VMware Tools. If you are running a graphical front-end to Linux there is a utility called VMware-Toolbox which allows further configuration. Lastly, ensure your Linux installation includes the tools required to use a C-compiler (such as gcc), as VMware Tools will need to compile the VMware drivers for your kernel.

Installing VMware Tools with the Redhat Package Management file

1. Logon as root to the Linux VM.
2. Use Control+Alt to regain control of the mouse and keyboard.
3. In Console, choose VM, VMware Tools Install.

   Note:

All this does is switch on the CD-ROM and point to the appropriate ISO file which contains the VMware Tools, located at /usr/lib/vmware/isoimages/linux.iso.

4. Next, mount this ISO file with mount –t iso9660 /dev/cdrom /mnt/cdrom as if it was a CD-ROM.

5. Execute the RPM file with the following command:

```
rpm -Uvh --nodeps VMwareTools-3.X.X-
XXXXX.i386.rpm
```

Note: The Meaning of RPM Switches

U stands for upgrade. Although this is a clean install, the same command could be used to upgrade VMware Tools to a newer version.

V is used to show for verbose information during the installation. H shows "hash marks" or status-bar like information which will tell us the progress of the installation.

Lastly, --nodeps forces an install regardless of software dependency errors. Here I have used a mixture of short switches which only need one – sign whereas longer friendly switches need two -- signs.

6. After the install process has completed you can use /usr/lib/vmware-config-tools.pl to configure the VMware Tools package.

Installing VMware Tools with a script

1. Logon as root to the Linux VM.

2. Use Control+Alt to regain control of the Mouse and Keyboard.

3. In Console, choose VM, VMware Tools Install.

Note:

All this does is switch on the CD-ROM and point to the appropriate ISO file which contains the VMware Tools. This is located at:

/usr/lib/vmware/isoimages/linux.iso

4. Next, mount this ISO file as if was a CD-ROM.

```
mount /dev/cdrom /mnt/cdrom
```

5. Copy the gziped version of the VMware Tools to the /tmp directory.

   ```
   cp /mnt/cdrom/*.gz /tmp
   ```

6. Unzip this gz file.

   ```
   tar –zxvf /tmp/vmware-linuz-tools.tar.gz
   ```

   Note:

   The z switch indicates that the tar command should uncompress the files. The x switch indicates that files should be extracted. The –v switch gives you a list of files being extracted.

7. Change into the vmware-linux-tools directory created by the unzip process, and run the installation script.

   ```
   cd /tmp/vmware-tools-distrib
   ./vmware-install.pl
   ```

8. Accept the default locations for the file copy.

   Note:

   The script will create directories for locations that do not exist currently.

9. Choose Yes, to allow the system to run the script. "/usr/bin/vmware-config-tools.pl"

   Note:

   This script configures VMware Tools for the first time. If you wish, you can run vmware-config-tools.pl with the -experimental flag, and this will allow you to enable the "Descheduled Time Accounting" driver as in Windows.

Adding Virtual Hardware

Hot Adding Virtual Disks

It's not possible to show you every possible VM configuration – a VM is simply too flexible to make this viable. However, I wish to give one popular example– that of giving a VM direct access to a SAN or iSCSI LUN. We will see other VM configurations later in this book; for example, running clustering services such as Microsoft or VERITAS Clustering Service within a VM which requires a special configuration.

If you wish to change the virtual hardware configuration (increasing CPUs, RAM or NICs) of your VM, in most cases you will have to power it down. However, in some guest operating systems you can "hot add" a virtual disk. This includes Windows XP Professional with Service Pack 2, Windows 2003, and many distributions of Linux.

Additionally, you may wish to allow your VM direct access to a SAN or iSCSI LUN, achieved by a special mapping file called a "Raw Device Mapping" (RDM). This metadata text file essentially "tells" the VM which LUN to access. Of course, the VM doesn't actually connect directly to the SAN or iSCSI system. Instead the VMkernel intercedes on its behalf using VMkernel drivers to access the SAN or iSCSI via the ESX host's physical HBA.

There are many reasons to do this. Firstly, while some companies are happy to store their data within the virtual disk format, more conservative companies prefer their data to be stored in the operating systems native file system. Secondly, you may have existing data held within NTFS, EXT3, or other propriety files systems to which you merely wish the VM to have access. This is quite common after a P2V process. Thirdly, RD's are required for some clustering scenarios – such as running a clustering service between two VMs on separate ESX hosts (referred to as a "cluster-across-boxes"). I will cover the various VM clustering options in Chapter 10 when I discuss high availability solutions. Lastly, you may wish to leverage your guest operating system's native disk and file system tools to carry out certain tasks. For example, Microsoft Diskpart tool allows you to "stretch" a NTFS partition to fill free space. This can be an advantageous feature if your SAN supports "stretching" a LUN to increase its size.

There are two compatibility modes when you create an RDM file – physical and virtual compatibility. Physical compatibility allows the VM to treat the raw LUN as if it was a physical machine and is primarily used in VM clustering scenarios. There are no special features or options with physical compatibility. Virtual compatibility, on the other hand, allows the VM to treat a raw LUN as if it was a virtual disk. It allows for advanced features such as different disk modes and VMware snapshot files.

RDM files have the extension of .vmdk just like virtual disks and can be stored alongside the VM's other files or a different datastore if you wish.

To add in a RDM on a running VM:

1. Right-click the VM and choose Edit Settings.
2. Click the Add button.
3. Choose Hard Disk from the list of devices.
4. Choose Mapped SAN LUN as the type of disk.
5. Select the LUN you wish to present to the VM from the list.
6. Choose to Store the RDM with a Virtual Machine.
7. Choose Physical compatibility.
8. Choose a SCSI node, for example SCSI 0:1.
9. Click Finish and Close the edit settings dialog box.

Note:

If you are running Windows you will need to rescan disks using Computer Management, Storage, right-click Disk Management, and choose Rescan Disks.

Hot Adding Virtual Disks to Linux

Adding hard disks to a Linux VM while it is powered is also possible. However, the process of rescan the virtual SCSI bus is not particularly easy. Fortunately there is a script which is freely available which allows rescan for new storage devices after adding virtual disk. The tool is a BASH shell script called scsi-

rescan.sh and was originally written by Kurt Garloff of Germany. You can find Kurt's script on the internet at his website under "Rescan SCSI Bus." http://www.garloff.de/kurt/index_e.html

Additionally you can find the script at Alex Mittell's website:

http://users.ox.ac.uk/~alexm/

Alex is a highly active member of the VMware Community Forums and also a member of the London VMware User Group, where he has given presentations. He's perhaps better known for his free Vi-3 backup utility called VISBU.

After adding the virtual disks to the Linux VM, download the scsi-rescan.tar.gz file, and extract it with the tar command.

```
tar –xvf scsi-rescan.tar.gz
```

Once the file has been extracted then execute the scsi-rescan.sh script. If it fails to execute try using the sh command.

```
sh scsi-rescsn.sh
```

This should run the script producing an output like so:

```
Host adapter 0 (mptspi) found.
Scanning hosts  0 channels 0 for
SCSI target IDs  0 1 2 3 4 5 6 7 , LUN's  0
Scanning for device 0 0 0 0 ...
OLD: Host: scsi0 Channel: 00 Id: 00 Lun: 00
      Vendor: VMware   Model: Virtual disk     Rev: 1.0
      Type:   Direct-Access                    ANSI SCSI
revision: 02
Scanning for device 0 0 1 0 ...
NEW: Host: scsi0 Channel: 00 Id: 01 Lun: 00
      Vendor: VMware   Model: Virtual disk     Rev: 1.0
      Type:   Direct-Access                    ANSI SCSI
revision: 02
1 new device(s) found.
0 device(s) removed.
```

Using the command fdisk – I will give you a list of all mounted, un-mounted, and un-partitioned drives like so:

```
Disk /dev/sda: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id
System
/dev/sda1   *           1          13      104391   83
Linux
/dev/sda2              14         196     1469947+  83
Linux
/dev/sda3             197         261      522112+  82
Linux swap / Solaris
Disk /dev/sdb: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk /dev/sdb doesn't contain a valid partition table
```

This means fdisk /dev/sdb could be used to partition and format the new virtual disk.

Using the Snapshot Manager

Currently, snapshots are applied to ALL virtual disks and RDMs in a VM. That would include both system drives that contain operating systems and application information – and also drives that store end-user data. Reverting a snapshot, which takes a VM "back in time," is therefore a potentially catastrophic event resulting in a loss of data. Some users have reported on the severe problems with committing snapshots above the 2GB size. There is a known issue if you have VMDK files within a VM that have the *same name* but are stored in *different VMFS datastores*. This problem is outlined in the VMware KB article 5096672. Lastly, some forum members have criticized the poor management of the Snapshot management file. This normally happens because of a failure to commit a snapshot file above the 2GB range. Due to some poor experience, the general opinion in the VMware Community is that while the snapshot's feature is generally a good one, it should be used sparingly until these issues have been properly resolved.

The new "Snapshot" feature replaces the old "redo" files of ESX 2.x. However, they have the same functionality and extra features. Snapshots allow you

to capture the state of a VM at a point in time (which includes both disk and memory states) and allows you to go back (revert to snapshot) to a needed point. A good example of using snapshots might just be a fundamental change to a VM where you are unsure of the consequences of your actions; for example, applying a new service pack. In this way we are using the snapshot to deliver a type of "undo" functionality. This has been available in ESX since version 2.x, but now we can have up to 32 levels of undo within a given VM, whereas in the previous release we were restricted to just 1 level. Snapshots can be created and deleted even when the VM is powered on. This is another improvement on ESX 2.x which previously forced us to power off the VM to then change the "disk mode" of a VM.

So in ESX 3.x we can make a change, click create snapshot, make another change, click make a snapshot, and so on we go. In this respect working with a VM is a bit like working with a file – saving as you go along – so you can go back to the last known good state of the file if something goes wrong.

Using snapshots during backup is also popular. When a VM has a snapshot applied, all the read and write events that would normally be sent to the virtual disk are actually sent to a "delta" file. Under normal operations (without a snapshot) the virtual disk is locked by the file system and cannot be manipulated. However, when snapshot is applied to a VM, the virtual disk is unlocked and can be copied to another location for backup purposes.

After creating the snapshot all the new changes in the disk and memory are actually going to a "differences," or delta, file. In this respect, when you "capture the state of a VM at a point in time" you are actually creating something more like a "bookmark" that you can use to return to a point in time.

Despite these really useful features, snapshots are not without their gotchas and best practices. Snapshots grow incrementally over time in blocks of 16MB. If you allow a VM to run on the "delta" file for a long period of time it could become quite large. The other concern, depending on how much disk I/O your VM generates, is the amount of free space required to continue running on the "delta" file. VMware recommend not allowing any snapshot to grow beyond 2GB in size for performance reasons.

Additionally, time sensitive operations could be disrupted by the "revert to snapshot" feature. Let's take an extreme example, such as taking a snapshot of VM *while it is copying a file* to another system through the network. Some hours or days later when you choose to revert to the snapshot this VM would still think it was copying a file. However, the destination system would still be in your time and the network file copy would fail. There are many systems that are time sensitive, especially authentication services like Microsoft Active Directory – so this is one to watch carefully.

CAUTION:

Using the snapshot feature incorrectly can result in loss of data. I recommend you take a test VM and play with this feature until you are entirely comfortable with it.

GOTCHA:

The Revert to Snapshot icon does *not* currently ask the operator "Are you sure?" Therefore it is incredibly easy to accidentally click and send the VM back in time!

TIP:

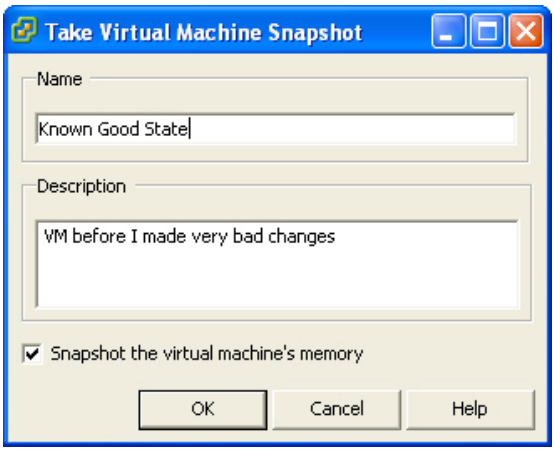
If you power down a VM first, it is much faster to take a snapshot because no memory contents need to be saved. When you revert the snapshot your VM is returned to its powered off state.

Creating a snapshot

1. Login to the VM.
2. In the menu choose **VM**, **Snapshot,** and **Take snapshot.**
3. **Type in a name and description** for the snapshot.

Figure 5.3 shows my dialog box. I am going to use this snapshot to demonstrate making a mistake and going back to a good state.

Figure 5.3

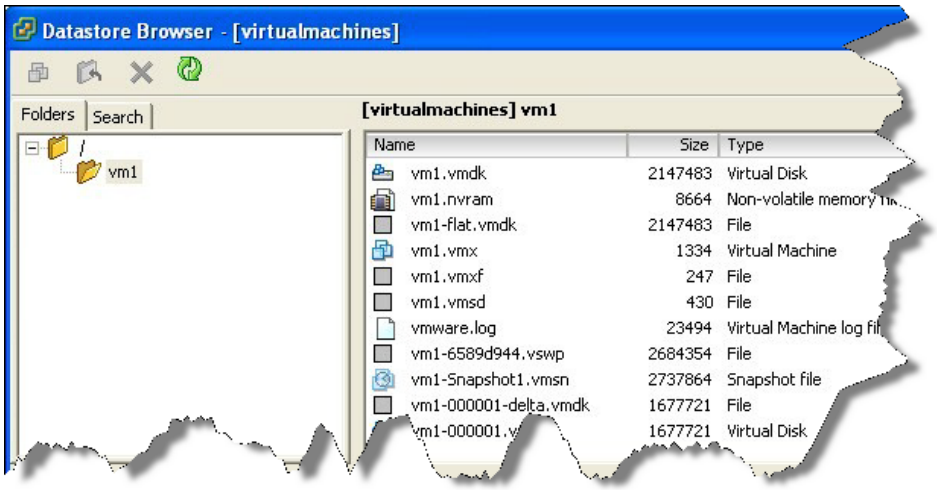


Note:

Now I make some changes I did not want. When I demonstrate this to customers in a Windows VM I tend to copy "Program Files" repeatedly. After doing this I then demonstrate the "Revert to snapshot" feature.

If you do this you might like to know you can see the "delta" file growing. To do this you can open SSH window on the ESX host in /vmfs/volumes or in the Vi Client using the "Browse Datastore" feature (as shown in Figure 5.4) by selecting your ESX host, clicking the "Summary" tab and then right-clicking a datastore in the resource pane and choosing "Browse Datastore."

Figure 5.4



Revert to a snapshot

There are two ways to control reverting to a snapshot. There is a silent method which does not give any prompts or warnings and the snapshot manager which assists in dealing with multiple snapshots.

Silent Method:

1. Choose **VM** in the menu.
2. Select **Snapshot,** and then **Revert to snapshot.**

   Note:

   As stated previously, because of the lack of prompts, be very careful with this option.

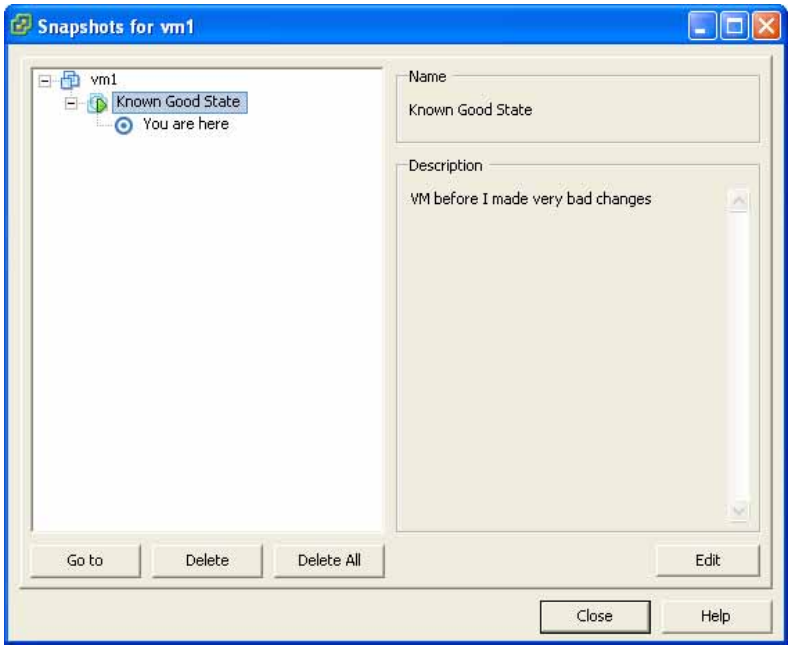
Snapshot Manager Method:

1. Choose **VM** in the menu.
2. Select **Snapshot** and then **Snapshot Manager.**

   Figure 5.5 shows my VM. Notice how the snapshot name and description assist me in remembering what state the VM will be before the revert process. The edit button allows me to change the name and description if this is unsuitable.

Selecting the snapshot I called "Known Good State" and clicking the "Go" button would allow me to move the VM back in time before I copied "Program Files." If I selected the "Delete" button it would take the contents of the "delta" file and *merge them into the VM's virtual disks* – finally deleting the delta file at the end.

Figure 5.5



3. When you click the "**Go To**" button you will be warned that your current state (with all my bad copies of program files for example) will be lost. This is not a problem as I don't wish to keep those changes anyway.

4. Choose **Yes.**

5. **Close the Snapshot Manager** window – in a short while your VM will be returned to its original state.

Warning:

Although we have gone back in time, or "Reverted to Snapshot," the snapshot feature is still engaged. If you re-enter the Snap Manager window you would still see the name of the snapshot. This can be very useful for repeated attempts at configuration processes where you have 5 or 6 steps and you are not sure of the correct procedure because of poor documentation from a software vendor.

It also mean that it is very easy to unknowingly leave a snapshot engaged. You can tell if you have a snapshot enabled from the Vi Client: ff the "Revert to Snapshot" button is dimmed, you are not using a snapshot; if it is colored then you are using a snapshot.

Deleting a snapshot

I've found that some of my customers struggle with this particular feature of the snapshots. Not least because we all feel uncomfortable with delete buttons – we fear we might lose the stuff we want to keep. It's worth saying very clearly that when you delete a snapshot using the snapshot manager you are not going to lose your changes in the "delta" file. Customers have more problem with the delete button's terminology rather than anything else.

Here's an analogy I use with my customers to help them conceptualize the dialog box. When you click "Create snapshot" it's like you have used a camera and taken a "photograph" of your VM at that time. Half-an-hour later you think you might make another change – so you take another "photograph" of the VM. When you click "Revert to snapshot" you are going back in time that half-an-hour; it is like a little bit of time-travel. When you choose "Delete snapshot" you are going back to those old "photographs" and deciding you no longer need them – because they are so old and out-of-date. Just because you delete an old "photograph" of VM it doesn't mean you will lose the current image you are using.

Here's what actually happens when you hit the delete button. VMware ESX server takes the contents of your snapshot and copies the data in the delta file into your virtual disk. Once the "delta" file has been merged with the virtual disk the delta file is then deleted. Some people prefer the old terminology of ESX 2.x which used the words "commit" to merge the file into the virtual disk, and "discard" to remove the file and revert back to last known-good state of the VM.

When I demonstrate this to customers I usually make a change I would normally wish to keep – such as password reset. I then delete the snapshot – and prove that my password change has taken affect. This helps to re-enforce in the minds of the customer that the delete button doesn't mean "lose my changes" but "keep my changes."

1. Within the VM make a change you wish to keep, such as a password reset.
2. Choose **VM** in the menu.
3. Select **Snapshot** and then **Snapshot Manager.**
4. Select your snapshot, and choose the **Delete** button.
5. Choose **Yes** to confirm you are happy to delete the snapshot.

   Note:

   In the VM you can prove the changes have been kept by logging in and out – and checking the password. If you browse the contents of the datastore you should find the delta files have been deleted but your changes have not been lost.

Changing Disk Modes

When you define a virtual disk you are asked to set its "disk mode." There are effectively 3 different modes:

- Non-independent mode (Default)
- Independent Mode with Persistent
- Independent Mode with non-persistent

Only the non-independent mode allows the snapshot feature. The persistent mode treats the virtual disk as a normal disk would be – any I/O is committed to the disk immediately, and snapshots are not allowed. Of course, you must still shutdown the guest operating system properly to flush the contents of memory to the disk. This is due to file-system caching, present in many modern operating systems.

On the other hand, the non-persistent mode marks the virtual disk to be volatile. Any changes made after this switch stops any I/O events from entering the disk. Every time you power the VM off and on your changes are lost. Some customers use this with test and development VMs or with training VMs that always need to be reset to a given state. What actually happens is changes accrue in a "delta" file, but at power off they are never merged into the virtual disks. Taken to the logical conclusion this could be very useful in a Virtual Desktop Infrastructure (VDI) environment. Imagine a situation where you only

have one VMDK file of Windows XP wasting valuable space on the SAN, and each use receives a "delta" version. At the end of the working day these VMs are powered off and reset to the golden state before the users made changes.

Lastly, changing disk modes does require the VM to be powered off.

1. Right-click the VM and choose **Shutdown down guest** from the menu.

   Note:

   The "Power off" does a hard power down akin to pulling out the power cord or hitting the reset button on some physical machines. It does not gracefully shutdown the VM. The "Shutdown down guest" option does require VMware Tools.

2. Choose to **Edit Settings** of the VM.
3. Select the **Hard Disk** in the list of devices.
4. Choose **Independent** and the **Persistent** mode.
5. Click **OK** to the Virtual Machine properties box.
6. **Power on the VM.**

Configuring VMware Tools

VMware Tools has a number of configurable options, especially in Windows. Most of these options are self-explanatory but it might be useful to discuss some of the most important ones.

Time Synchronization

The most common configuration for time synchronization is to enable the Network Time Protocol (NTP) service on the ESX host. The NTP service that provides accurate time to ESX is either on your own network or on the internet. Using VMware Tools the VM synchronizes its time with the ESX.

In VMware Tools for Windows and for other guest operating systems this is not a default. The reason being is many guest operating systems have their own

time synchronization feature which would conflict with VMware Tools; in the case of Windows this is the service "Windows Time." You cannot have two time synchronization services within the same machine – the services would conflict with each other, and the VM would not be a trusted source for time. To use the time synchronization feature from VMware Tools you must disable these guest operating systems methods first.

This VMware Tools version of time synchronization happens once every minute and is not currently configurable. For this reason, some time sensitive VMs might still need their time set from systems that update their time at a more frequent interval.

To enable a Windows VM to use VMware Tools time synchronization, use the VMware Tools icon in the taskbar tray area. In other guest operating systems we enabled it by editing the .vmx file of the VM.

Enabling VMware Tools time synchronization in Windows

1. In Administrative Tools and Services
2. double-click the Windows Time service, and choose Stop.
3. From the Start-up Type pull-down list choose Disabled.
4. Close the Service console.
5. Double-click the VMware Tools icon in the tray, and enable "Time synchronization between the virtual machine and the console operating system."

   Note:

   Special considerations must be followed if your VM is running an Active Directory and is a domain controller. The VMware KB article 1318 outlines this.

   "*If you use a virtual machine as a primary domain controller for a Windows network, the primary domain controller must run the Windows Time service as a time server, to provide time to secondary domain controllers and other hosts on the network. However, that primary domain controller does not need to use the Windows Time service as a client to receive time synchronization input for its*

own clock. You can still use VMware Tools to synchronize the virtual machine's clock while running the Windows Time service in a server-only mode."

This is done by engaging the Windows Registry option called "NoSync."

For this information and more detailed explanation of time inside a virtual machine consult the following VMware Documents:

VMware Time Sync and Windows Time Service:

http://kb.vmware.com/selfservice/microsites/search.do?cmd=displayKC&externalId=1318

Timekeeping in VMware Virtual Machines (PDF):

www.vmware.com/pdf/vmware_timekeeping.pdf

Enabling VMware Tools time synchronization in other guest operating systems

1. Shutdown your VM.
2. Open a SSH session on your ESX host, elevate your rights to root using the su – command.
3. Use nano or your preferred text editor to open the VM's VMX file. An example follows.

   nano -w /vmfs/volumes/virtualmachines/vm2/vm2.vmx.
4. Scroll to the end of the file, and find the tools.syncTime = "FALSE" option, modify this to read

   tools.syncTime = "TRUE"
5. Save the file, and Exit your text editor.
6. Power on your VM.

Configuring Scripts

As mentioned before, we can have scripts executed when a VM's power status changes. There are a couple of examples of configuration power-state scripts. Firstly, it is sometimes quicker to reboot operating systems like Windows by

stopping services before the calling to reboot the VM. Running application services like Microsoft Exchange where you could use a .bat file with Microsoft net stop command to stop services is a good example.

Secondly, there are sometime annoyances like dialog boxes that stop successful reboots or shutdowns; rather than having to logon to the VM and deal with these prompts you could script them away. In my work, I deal extensively with Microsoft Terminal Services and Citrix MetaFrame. If an administrator uses the "Restart Guest" option in the Vi client, they could find that pop-ups appear within Windows dialog boxes (Figure 5.6 and 5.7 illustrate this). At a request to reboot or shutdown the VM triggered from the Vi Client, the request merely times out if there is no-one to answer these dialog boxes.

Figure 5.6

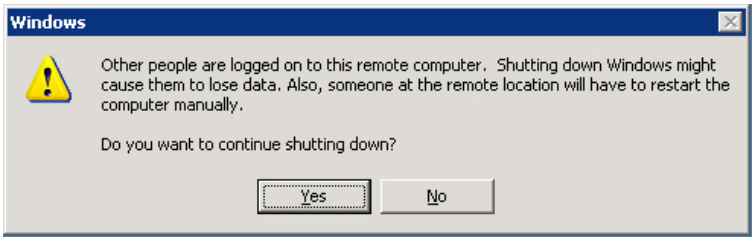
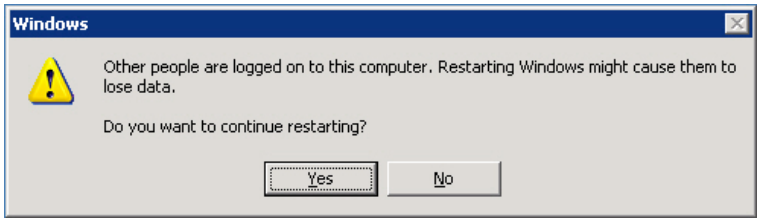


Figure 5.7



To fix this problem I used Microsoft tsshutdn command with the power-down script.

1. Login to your VM.
2. Browse to **C:\Program Files\VMware\VMware Tools**.
3. **Right-click** the **poweroff-vm-default.bat** file, Choose **Properties** and remove the "**read only**" attributes.
4. **Right-Click** and **Edit the file**.

5. Add to the bat file:

   tsshutdn 0 /reboot /delay:300

   Note:

   tsshutdn has many options. The two 0 values stop any warnings or delays and starts the shutdown immediately. If I used tshutdn 120 /powerdown /delay:30 this would give the users 120 seconds to log off, and then power down would begin 30 seconds after all log-offs had completed. The messages go to all users whether you are using the Microsoft RDP or Citrix ICA protocol.

   GOTCHA:

   Shutdown and reboot guests use the *same* script in VMware Tools. This means if you used the above workaround – and signaled the VM to shutdown – it would in fact reboot. The only way to shut-down the guest would be to login and do a manual shutdown within Windows.

Using the Shrink Feature

In ESX 3 the shrink feature has been depreciated – in fact, it's no longer supported by VMware in a ESX 3 VM. This is a shame because it is actually a useful feature. Shrink optimizes a disk before exporting (copying) it to another storage system by deleting deleted files. As you might know most guest operating systems do not actually delete files physically from either a physical or a virtual disk. Files are marked for deletion in the file system database and then are over-written by new files. The downside for us is that when we come to copy a virtual disk elsewhere, say prior to a backup, we get both our real data and our deleted data. Shrink used to write out the deleted files with zero values thus reducing the overall size of the disk – hence the term "shrink."

However, all is not lost. Many community forum members use a tool called sdelete from what used to be sysinternals.com. Microsoft purchased the website and its tools in July 2006. You will now find them re-named as Windows Internals and sdelete is listed under "File and Disk" utilities.

http://www.microsoft.com/technet/sysinternals/default.mspx

There are plenty of secure delete style tools available for other guest operating systems such as Linux, Solaris, and Novell Netware.

Auto-Start and Stop VMs

ESX has the ability to gracefully power off and on your VMs if choose to do a shutdown or reboot of a ESX host. It is very easy to configure:

1. Select your ESX host, and Choose the Configuration Tab.
2. In the "Software Pane" select Virtual Machine Startup/Shutdown.
3. In the top right-hand corner select Properties…
4. In the dialog box, under System Settings enable "Allow virtual machines to start and stop automatically with the system."
5. Under shutdown action drop-down option Choose "Guest Shutdown."

   Note:

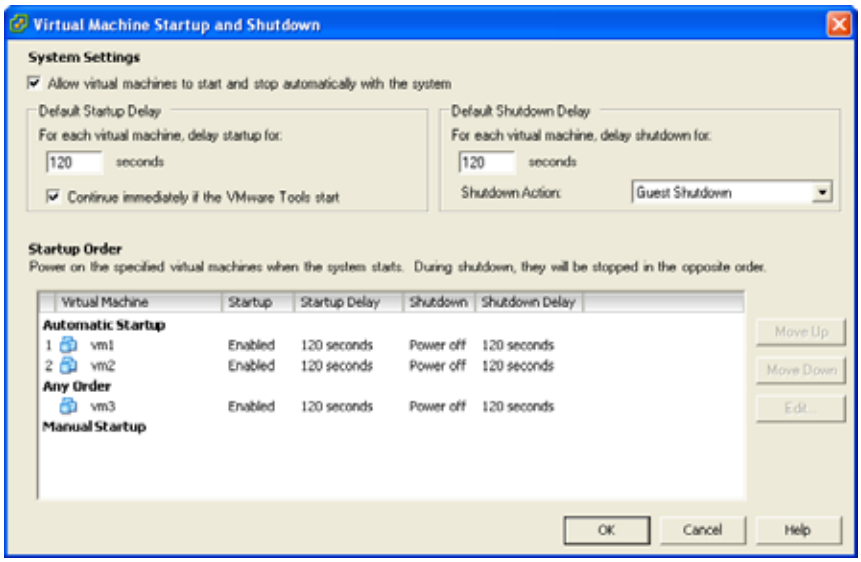
   "Guest Shutdown" requires the use of VMware Tools. This sends a signal to the VM to begin its shutdown process.

   You can use the up and down buttons to control the order of shutdown and start-up for VMs that share service dependencies.

Additionally, you can modify the start-up and shutdown interval used between VMs to reflect the fact that some VMs services will take longer to start than others. As an option, you can ask the system to over-ride the start-up daily by monitoring for the start VMware Tools, which may begin sooner than the 120 second default value. Lastly, the edit button allows you to set individual settings for each VMware over-ride to the global "System Settings."

Figure 5.8 shows the Virtual Machine Startup and Shutdown dialog box.

Figure 5.8



GOTCHA:

While it is possible for ESX to stop and start VMs whenever you reboot an ESX host, it is more useful in an environment that does *not* have features such as VMotion, VMware DRS, and VMware HA. If you have these features in place then the auto-start and stop feature will not do much for you. In fact, this feature "breaks" as soon as a VM is moved from one ESX host to another by either VMotion, DRS, or HA. This is not a bug, but design. After all, as soon as we have VMotion, DRS, and HA we start to care less about where our VM runs, as long as it does run. If you manually VMotion a VM away from the ESX host that has this list configured and then return back to the ESX, it is dropped in the default location of manual start-up.

P2V of Physical Machine with VMware Converter

In fairness, P2V should be a book in its own right and is really beyond the scope of this chapter. However, I felt any chapter about creating VMs that did not at least give a brief note of converting physical machine to virtual machine, would look rather remiss in a book that extols the virtues of vitalizing existing physical environments!

In the real world, many companies opt for what is termed a "P2V Jumpstart." This is where experienced consultants from your P2V vendor visit your organization for about a week. During that week they introduce and set up the software required – and assist you in your first few P2V events. This workshop approach works better than, say, conventional classroom training which tends to be rather unrealistic. After all, there are many different types of servers, operating systems, and applications. You really need to know your hardware, operating system, application software, and your environment before embarking on the P2V process.

You're not restricted to using VMware's software in this process. The leading vendors in the third-party market are:

- VMware Converter (nee VMware P2V 2.x)
- PlateSpin PowerConvert
- LeoStream P2V

Additionally, some hardware vendors like Dell, IBM, and HP have gotten in on the act, offering their own tools for converting the physical systems into VMware VMs. Another alternative is to check out so-called "free" P2V solutions –however, these free "solutions" do not ship with any warranty or commercial support. Additionally, they are unlikely to have fancy post-configuration P2V features. Some interesting tools include:

- Ultimate-P2V (www.rtfm-ed.co.uk)
- MOA Project (www.sanbarrow.com)
- Easy P2V (www.ezp2v.nett)

You might wish to investigate how these various tools actually achieve the "cloning" process. Some vendors install an agent into the existing physical machine which then allows it to be visible to management console used for cloning. The advantage is that you can remotely convert the physical machine while it is powered on, which can be important because of uptime challenges that P2V inherently introduces. The disadvantage is that you have "altered" the original physical host. Many organizations have an ideological problem with this approach – they argue this could affect fail-back procedures and prefer the physical machine to be "closed" during the conversion process.

Other P2V solutions make the physical server from a boot from a CD and duplicate the server. This is advantageous because you can be 100% sure that every file will be copied as there are no open files, and no changes are made to the original server. The disadvantages are server downtime and the possibility that the vendor's boot CD will not recognize the hardware (critically NIC and storage controller). The best P2V vendors will offer a combination of both – like VMware Converter.

Lastly, many of these tools are geared up for Windows P2V events although some of them do offer Linux based conversions, too. So if you're working in a heterogeneous operating system environment you might want to research the guest operating systems supported. VMware Convertor is Windows application which is installed to your management PC. It is currently limited to "experimental" support for Linux.

In this section I am going to look at a very small part of the P2V process – the conversion software – and outline some very simple "clean-up" routines also. I will be using VMware Converter. VMware Converter ships in two formats – "Starter" and "Enterprise." The Starter Edition is agent-based and is free. The Enterprise Edition can use an agent or boot CD. Both the Agent and Boot CD possess the same user interface so if you have access to both there isn't any learning curve.

There are some other important limitations of Starter about which you should be aware. There are two ways of using VMware Converter with Vi-3.

1. Installing the full VMware Converter software into the physical machine.
2. Triggering the install of the VMware Converter agent and managing it remotely with a management PC.

The Starter Edition only supports method 1 whereas the Enterprise Edition supports both. This means if you use starter you have to install about 15MB of software to your physical machine and be at the physical machine to do the conversion. This said, RDP and ILO connections are unaffected.

Lastly, although I am emphasizing the physical to virtual functionality of converter, you should know that it has lots of other cool features such as:

- Converts VMware VMs across multiple VMware platforms – and therefore is compatible with ESX 3.x, 2.x, Workstation 4 and higher, Player, and Server. It is also backwards compatible with versions GSX (since re-marketed as VMware Server).
- Converter third party formats like Symantec Backup Exec Recovery, Norton Ghost, Microsoft Virtual Server, and Microsoft Virtual PC.

GOTCHA:

After a P2V process has completed, some editions of Windows will need reactivating. Windows will see the new virtual disk as a brand new hard drive, and the GUID associations with the old hard drive will be reset.

VMware Converter with the Agent (Enterprise Mode)

Before you begin verify you can login to the physical server with administrator credentials and ensure you have no mapped drives or other network connections to the physical server on your management PC.

1. **Download** VMware Converter.
2. **Install VMware Converter** to your **management PC.**
3. **Run VMware Converter,** and click the Licensing Button.
4. **Browse to your license file.**

   Note:

   If you are using the Starter Edition, install the VMware Converter product to the physical machine.

   You can run Enterprise Edition of VMware Converter in a VM.
5. Click the **Import Machine** button.
6. **Step 1**: Choose a source.
7. Select the option **Physical Computer.**

8. Choose the option for a **Remote Machine.**

   TIP:

   If you cannot get communication or authentication working you can always resort to installing the whole of VMware Converter to the physical server and use the "local machine" option instead.

   Note:

   Type in the name or IP address of the physical server. Then type in the Administrative Credentials for the Physical Machine. These must be expressed in the format of DOMAIN\Username if you do not know the local administrator account or password.

   Once VMware Converter has connected to the physical machine, the converter will install an agent to the physical machine. At the end of the conversion process the agent can be automatically uninstalled or manually uninstalled- it is it up to you. The agent installs itself as service called VMware Converter Service in Windows.

9. Select the option **Automatically uninstall the files when import succeeds.**

10. **Select the Physical Volumes.**

    Note:

    You need not necessarily copy your data – you can just select your OS partition. Additionally, you can choose to resize disks as well.

11. **Step 2**: Choose a Destination.

12. Choose **VMware ESX server or VirtualCenter virtual machine.**

13. **Enter the name** of your **ESX Host** or **VirtualCenter server and user account details.**

14. **Type in the name of the new VM**, and **Select the folder** in VirtualCenter to hold the VM.

15. **Select an ESX host** location for your VM.

    Note:

You must initially select the ESX host, not a DRS or HA cluster label. Once the VM is powered on, as long as the VM fulfils the requirements of DRS and HA, they will manage where the VM runs.

16. **Select a VMFS or NAS datastore.**
17. Select which **Network Port Group you wish to use.**

    Note:

    I would recommend initially using an internal switch or having the virtual NIC disconnected to avoid any potential IP or NETBIOS name conflicts.

18. Enable **Install VMware Tools.**
19. Click **Next** and **Finish.**

    Note:

    You can watch the status of you conversion from the converter windows in a percentage. Additionally, the "Task Progress" tab will give you an overview of the steps the converter is completing.

    Note:

    I would recommend choosing "NO" to the option of powering on the P2V'd VM at the end. There is some clean-up and post-configuration we can do from the Vi Client before the first power on.

VMware Converter with the "Cold-Clone" Boot CD (Enterprise Mode)

The Enterprise edition of VMware Converter also comes with the option to download a boot CD. This allows you reboot a physical server and clone the disk while the system is offline. The boot CD is actually a modified version of Microsoft WinPE environment. Previous editions have used a Debian CD and then later the Knoppix Live CD. I don't know what prompted VMware to move in this direction but I think the reason was threefold:

- It is substantially easier to add additional drivers for networking and storage.

- WinPE's competitor is the highly popular BartPE – however, to run BartPE according to Microsoft, you should really purchase a license for Windows XP or Windows 2003. Many people do, not which is very naughty of them as it upsets Microsoft a great deal. WinPE has the advantage that VMware can distribute it under a legally water-tight license agreement – to customers who perhaps don't even use Windows.
- It allowed VMware developers to write a very easy and intuitive UI, consistent both in Agent and Cold-Clone modes.

GOTCHA:

Firstly, VMware Converter supports all the flavors of Windows – and is happy if the disk is basic or dynamic. It will not convert volumes configured with Microsoft's software implementation of RAID. Secondly, you need at least 264MB of RAM for cold-cloning to work. If your memory size on the physical system is more than 364MB, the boot CD will create a RAM drive which improves the performance of the CD.

1. **Download the ISO** from VMware's website.
2. **Burn to a CD using your burner software** (if your server supports ILO or RAC boards with Virtual Media you could just use the ISO file as is).
3. At the prompt **press any key to boot from the CD…**
4. At the dialog box choose **Yes** to "**Would you like to update network parameters at this time.**"

   Note:

   Confirm your DHCP server has leased the boot CD an IP address. If you don't have access to a DHCP server, input your static configuration.
5. Click **Import Machine.**
6. **Step 1:** Import Preparation.
7. Choose to **Select volumes, and resize to save or add space.**
8. Select the **Boot Partition of your physical server.**

9. **Step 2**: Select a Destination.
10. Choose **VMware ESX Server or VirtualCenter virtual machines.**
11. **Provide login name and credentials** to access **your ESX host or VirtualCenter Server.**
12. **Type in a name for your new VM,** and select a location.
13. **Select an ESX host** to run the VM.

    Note:

    Remember, as with agent-driven conversions, you must initially select the ESX host, not a DRS or HA cluster label.
14. **Select a datastore location** for the VM.
15. **Choose a network port group** for the VM.
16. Allow the system to **Install VMware Tools.**

    Note:

    When the conversion is over select File and Exit in the Converter.

Post-Configuration Changes

After any P2V conversion there is a significant amount of clean-up work that needs to be completed. Here's a brief check-list of the kind of tasks you may need to consider:

- Remove Stale Devices.
    - Shutdown the VM.
    - Edit the Settings of the VM.
    - Remove legacy devices like Serial Ports and Parallel Ports.
- Remove Stale Software such as:
    - Hardware Drivers (Graphics, Sound, NIC's, RAID Controller). Doing this first reduces the time spent on remov-

ing stale hardware, as de-installing drivers sometimes removes the references in hardware management tools.

- Remote Access Software such as VNC
- Hardware Agents such as HP Insight Manager, IBM Director and Dell OpenManage

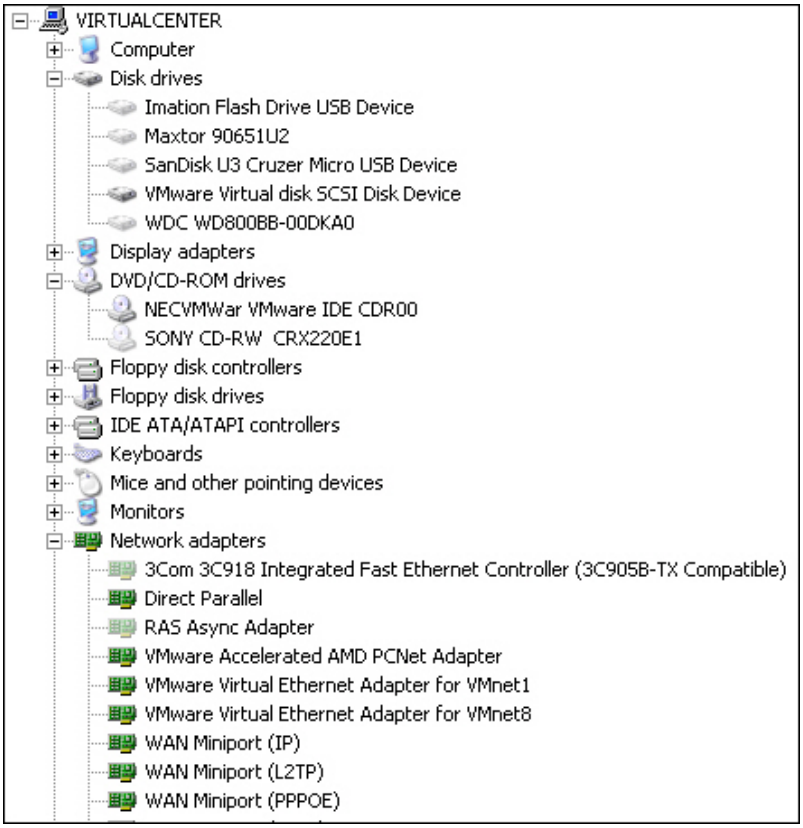
- Remove Stale Hardware with Device Manager.
  - Run Device Manager with this batch file:

    ```
    @echo off

    cls

    set devmgr_show_nonpresent_devices=1

    start devmgmt.msc
    ```

  - In Device Manager, change the view to Show Hidden Device (Windows 2000's Add Hardware Devices has a similar option).
- This truly shows you the old hardware – which you can right-click and choose Uninstall.

  Figure 5.9 shows the stale hardware of P2V's VM. You can see the old hard-drives (a Western Digital and a Maxtor) together with various USB sticks that have inserted into the physical server. We can also see an old Sony CD-RW drive – and last, the old 3COM network card.

Figure 5.9



- **Reconfigure Networking**

  After the P2V your old network card has gone, and your new network card(s) from VMware will have no IP settings.

As you can tell the post-configuration process is not insignificant. Much of this process could actually be scripted. RTFM Education's website has a white paper and some sample scripts that have been developed to address this issue. An interesting one uses Microsoft's DevCon utility to compare the Virtual Machine to the Physical Machine – and automatically remove the stale hardware that appears in Device Manager. Research using the ESX 2.x. platform showed the script could remove about 60 to 80 unwanted devices in most Windows environments.

http://www.rtfm-ed.co.uk/?page_id=8

Removing, Adding and Deleting a VM

It is possible to remove a VM currently registered on ESX host and listed in VirtualCenter and add it to another ESX host in a different VirtualCenter environment. The important thing to note is that the only requirement is for shared storage. Effectively this achieves a manual moving of a VM from one VirtualCenter environment to another. If you have re-installed an ESX host or VirtualCenter - you might need to add VMs to the host in order to power them on.

Removing a VM does not delete the files that make up a VM, it merely removes the VM from the ESX host and VirtualCenter lists. A similar concept exists when you remove virtual disks inside a VM which is called "Remove from Virtual Machine" and "Remove and delete files from virtual disk."

To Remove a VM

1. Power off the **VM.**
2. Right-click a **VM**, and Choose **Remove from Inventory.**
3. Choose **Yes.**

To Add a VM

1. Select an ESX host.
2. From the Summary page, right-click the datastore where the VM resides and choose Browse Datastore.
3. Locate the VM's VMX file which is held within its directory.
4. Right-click the VMX file, and choose Add to Inventory.
5. In the Wizard, name and choose a folder location for your VM, and next your way through the remaining dialog boxes.

GOTCHA:

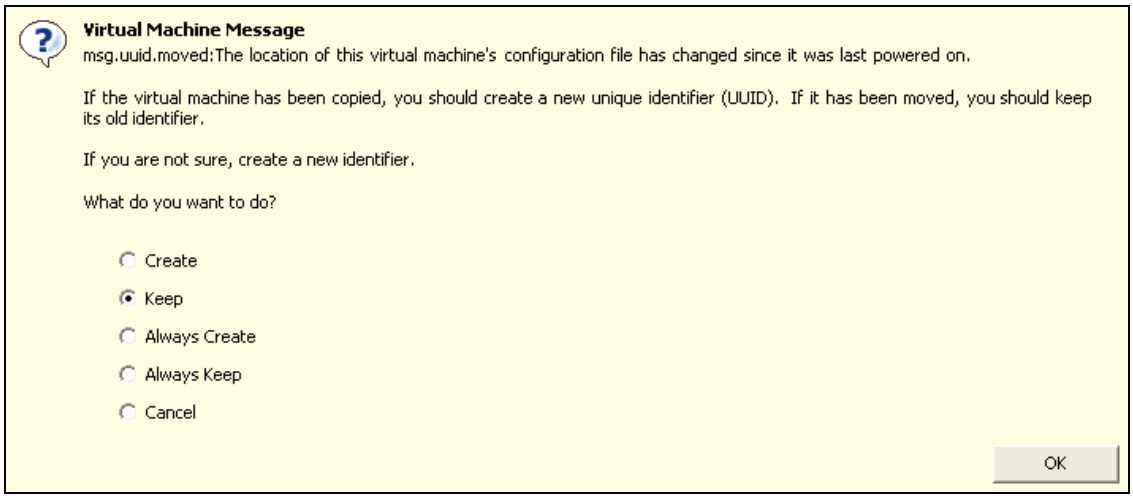
Adding and removing a VM is a relatively simple task – but beware of doing this from one ESX host to another. In other words, unregistering a VM from one ESX host and registering it to a different ESX host. When the VM is powered a

prompt will appear asking what you would like to do with a change in the VM's UUID. The UUID stands for the "Universal Unique ID." It's often used by management systems to track pieces of hardware separately from the OS it runs. Physical servers have a UUID and this allows us to wipe Windows from a physical server and install Linux – but still have the management system recognize it as the original piece of hardware. In simple terms, the UUID is hardware identifier which has no dependencies on the operating system installed. VMs also have UUID value which is held in the VMX file and is generated from the real UUID of the ESX host.

In most cases it is best to choose "Keep" to retain the VM's identity in management systems of this type. If you were *moving a VM you should create a new UUID*. This means a VM manual moved retains its original identity or UUID. If you were manually *copying a VM*, *you should create a new UUID.* What we need to avoid is two VMs with the same UUID as this would cause problems in management systems that use the UUID.

Figure 5.10 shows the UUID prompt that occurs in this scenario. If you move a VM using VirtualCenter either powered on (VMotion/Hot Migration) or powered off (Cold Migration) the UUID is unaffected.

Figure 5.10



Deleting a VM

Deleting a VM is a permanent operation. There is no undo button or recycle bin in ESX. If you accidentally delete a VM it is gone for good. Your only resort would be restoring it using your backup strategy. Similarly, there are no triple "Are you sure?" style dialog boxes or please confirm with a 4-digit pin number. You are asked once "Are you sure?."

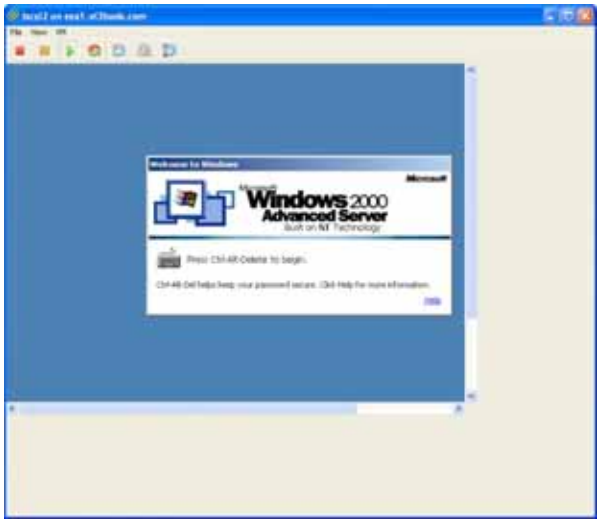
1. Right-click a **VM.**
2. Choose **Delete from Disk** – and read the dialog box!
3. Choose **No**, if you are at all unsure!

VM Console Resolution Annoyance

Lastly there is an occasional annoyance with the VM console window - in fact you may have already experienced this issue in your use of the Vi Client with a VM. It usually happens during the power on and boot process as the guest operating system returns different resolutions as it loads its graphics drivers.

The problem looks like Figure 5.11 below:

Figure 5.11



Generally, there is no "fix" to this problem at this moment. In my experience I've found that if you first "maximize" and then "restore" the VM console window like above, and then in the menu choose **View**, **Fit to Window,** this fixes the problem. If I don't first maximize and then restore the window I find that VM console creates a window which is 640x480 and sized to where the scroll bars are the above screen shot. This can be troublesome as it appears that the only way to return it to 800x600 or higher is by configuring it in the guest operating system display options. In the case of Windows this means opening the "Display Options" from the desktop.

Conclusion

As you have seen, VMs are infinitely more flexible than physical machines. Virtualization isn't just about taking your existing physical servers and converting them to VMs, although we did indeed touch on that subject in this chapter. It's about liberating you from the constraints of physical hardware.