



An IT pro's overview of Windows 7

An IT pro's overview of Windows 7

Windows 7 features reference guide	3
Explore the Problem Steps Recorder in Windows 7	14
Take advantage of tools to monitor memory usage in Windows 7	17
Secure your USB drives with BitLocker To Go for Windows 7	21
Windows 7: Mobile Data Protection with Bitlocker To Go	24
PowerShell 2.0 - Community Technology Preview (CTP) 3 released	27
10 cool things you can do with Windows PowerShell	29
Determine if your hardware can support Windows XP Mode in Windows 7	33
10 reasons why Windows 7's XP Mode is a big deal	36
Configure a USB flash drive to be a Windows 7 installation platform	38
Windows 7 now allows you to boot a native OS from a VHD file	41

Windows 7 features reference guide

The release of Windows 7 will be a big day for many IT shops. If your organization is still determining whether to migrate to the new operating system, this overview of tools and features, available in the RTM version of Windows 7, serves as a good reference.

Note: Not all of the features mentioned here are available within every version of Windows 7.

The table below outlines the versions of Windows 7 as well as the feature set contained within each version.

Table 1

Windows 7 Starter *	Supports the latest hardware and software, jump lists, better reliability, home group support, more usable taskbar
Windows 7 Home Basic **	Live thumbnail preview, advanced networking (ad-hoc wireless and Internet connection sharing), Windows mobility center
Windows 7 Home Premium	Advanced navigation, Aero glass, Easy networking for sharing devices across PCs, Windows Media Center, better media format support, multi-touch
Windows 7 Professional	Domain Join for simple networking, Encrypting File System, Location Aware Printing, Remote Desktop Host
Windows 7 Enterprise and Ultimate	Bit Locker, Bit Locker to go, Direct Access, Branch Cache, App Locker, Multiple Language Support, Boot from VHD

* *Windows starter edition will be limitedly available*

** *Windows Home Basic will be available in emerging markets only*

Note: Features marked in the above table are cumulative as the features increase. Windows Starter edition features are included in Home Basic and so on as the edition (and cost) increases.

Each of the flavors of Windows 7 gains features as you move down the list, this time around, there will be no lost features upgrading from one version to the next. If you start with Home Basic and upgrade to Professional, the original feature set will be there after the upgrade.

With Windows Vista, Microsoft introduced Anytime Upgrade, which allowed any user to upgrade their version of Windows Vista. This was a good concept, but in my opinion this ability was missing things. Some features were missing and the process was not as smooth as I would expect. Anytime Upgrade in Windows 7 is a better process and much easier on the user. Suppose I have Windows 7 Home Basic on my PC and I decide I want to go to Home Premium. I can do this using Windows Anytime Upgrade just as in Vista; however, the existing Windows 7 installation contains all the bits for Windows 7 when installed. Upgrading changes the product key for windows and enables more features, making the process much smoother and requiring no loss of features.

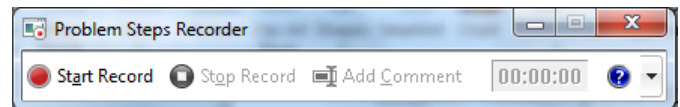
Administrative and support tools

Windows 7 includes improved tools to help IT support work with users and their computers to collect the necessary data to speed up the troubleshooting process.

Problem Steps Recorder

One of Windows 7's most significant additions is the Problem Steps Recorder, which allows a user to record the process of performing the task that caused an error. When a failure is reported by a user, typically I will ask them to repeat the steps they took to see if the failure happens again. In Windows 7, a user can turn on the Problem Steps Recorder before performing the task again; this will capture the process and zip the screen shot on an e-mail attachment to send to support. To get to the Problem Steps Recorder you can search for it from the start menu or the Control Panel. Entering "PSR" in either place will locate the tool for you. **Figure A** shows the Problem Steps Recorder in Windows 7.

Figure A



Problem Steps Recorder

Related TechRepublic content: [Explore the Problem Steps Recorder in Windows 7](#)

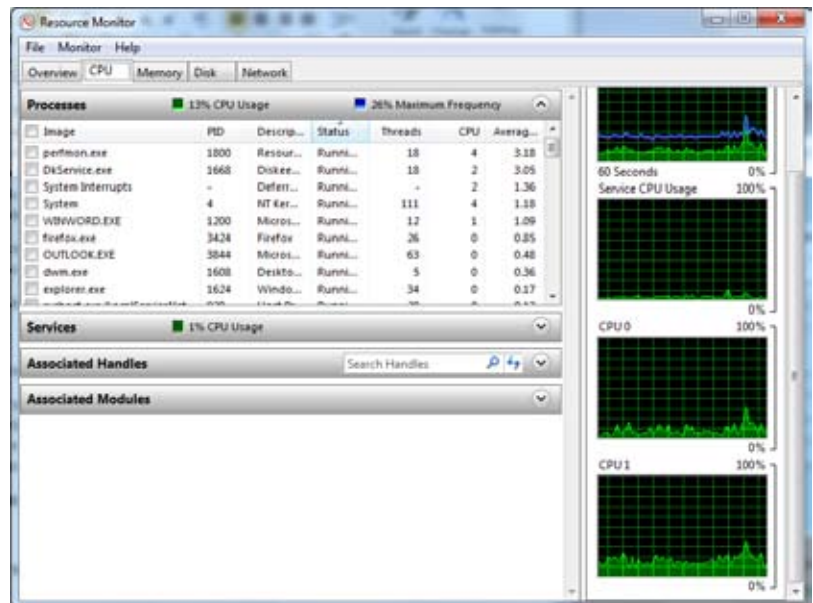
Resource Monitor

Resource Monitor, which replaces Performance Monitor, shows the computer's disk, memory, and CPU performance. Access to Resource Monitor is found in Accessories -> System Tools on the Start Menu. It is also available in Administrative Tools in the control panel (labeled Performance Monitor) or by searching for Resource Monitor on the Start Menu. Multiple Core CPUs are displayed as multiple CPUs within Resource Monitor. **Figure B** shows Resource Monitor on a computer with two cores.

Resource Monitor is a local use tool; to use Resource Monitor on a PC, you will need to be on the machine that needs to access it.

Related TechRepublic content: [Take advantage of tools to monitor memory usage in Windows 7](#)

Figure B



Resource Monitor CPU activity

Windows Firewall With Advanced Security

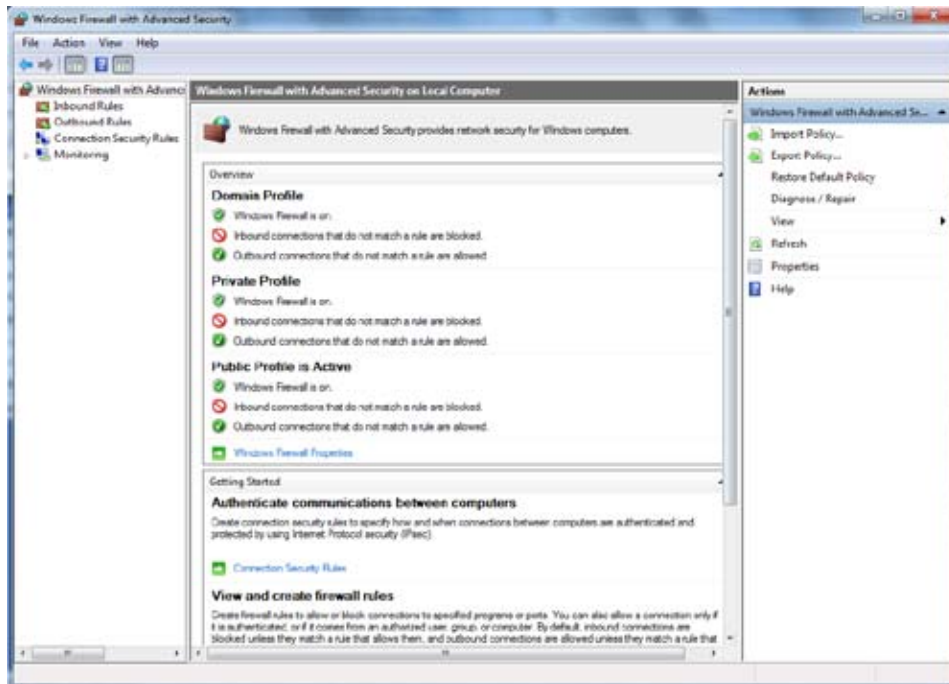
Windows XP introduced an inbound firewall to help prevent incoming connections to a computer. IT Pros could create exceptions to allow users access to resources that may not have been allowed by the default configuration. To review the Windows Firewall settings in Windows 7, navigate to Administrative Tools in the Control Panel to open the Firewall application. In Windows 7, the Windows Firewall is improved; it supports inbound and outbound connection filtering, and it is enabled by default.

The Windows Firewall in Windows 7 also supports policies, which are sets of rules for inbound and outbound connections. For instance, suppose the IT department wants to create a policy for the Windows Firewall containing rules to allow inbound and outbound connections requested by the Live Meeting client. (Live Meeting's client modifies this on installation to be allowed.) Once the rules are created on one system, the rules can be exported and shared via multiple computers. This expedites the use of policies in the Windows Firewall. Group Policy can be used to deploy Windows Firewall policies to systems within your environment running Windows 7. This will ease the administration of firewall settings for Windows 7 by ensuring they are applied across the organization. **Figure C** shows Windows Firewall With Advanced Security.

Many organizations will not change the Windows Firewall default settings unless they have a number of remote users, because they will have a hardware firewall configured to protect the entire corporate network.

Related TechNet Webcast: [Windows 7 Enhanced Security and Control](#)

Figure C



Windows Firewall With Advanced Security

Print Management

Print Management makes the tedious task of managing printers and documents easier. Print Management is available from the Administrative Tools option in the Control Panel. - or - Print Management (Control Panel | Administrative Tools | Print Management) makes the tedious task of managing printers and documents easier.

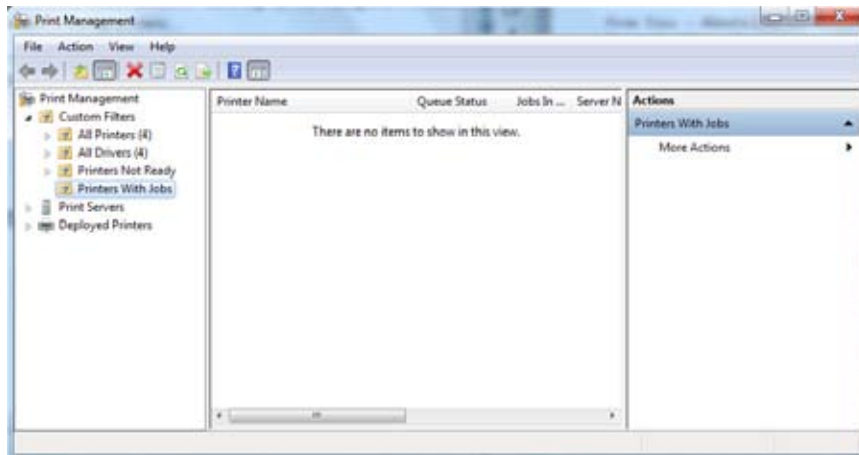
Print Management also provides a great deal of information for administrators to use to troubleshoot and work with Print services. (**Note:** Don't confuse Print Management with the Printers And Drivers option on the Start menu.)

The Print Management console shows printers with jobs in a group, which will help you find stuck print jobs in a single view. Other Print Management features include visibility of print servers and drivers and visibility of any printers connected to a computer (either locally or via the network).

In the More Actions section of Print Management, you can set notification options that can alert you via e-mail when there are printer issues. You can also schedule a script to run if the notification is triggered. Occasionally, I need to restart the print spooler service on a print service; instead of touching the print server, I could use More Actions to schedule a script to restart the service and notify me that this has occurred.

Figure D shows the Print Management console.

Figure D

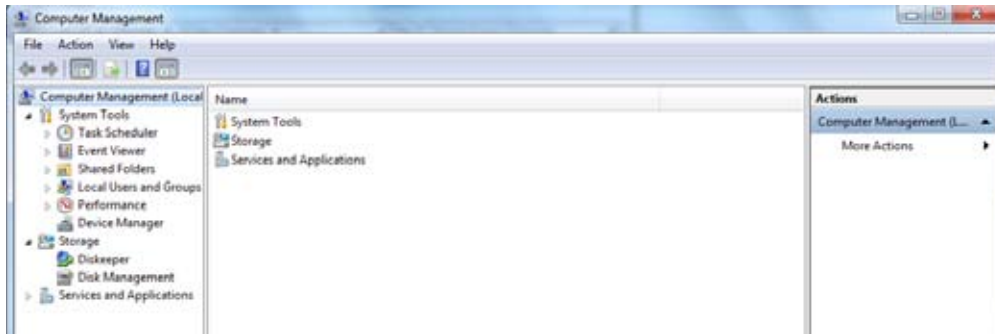


Print Management

Computer Management

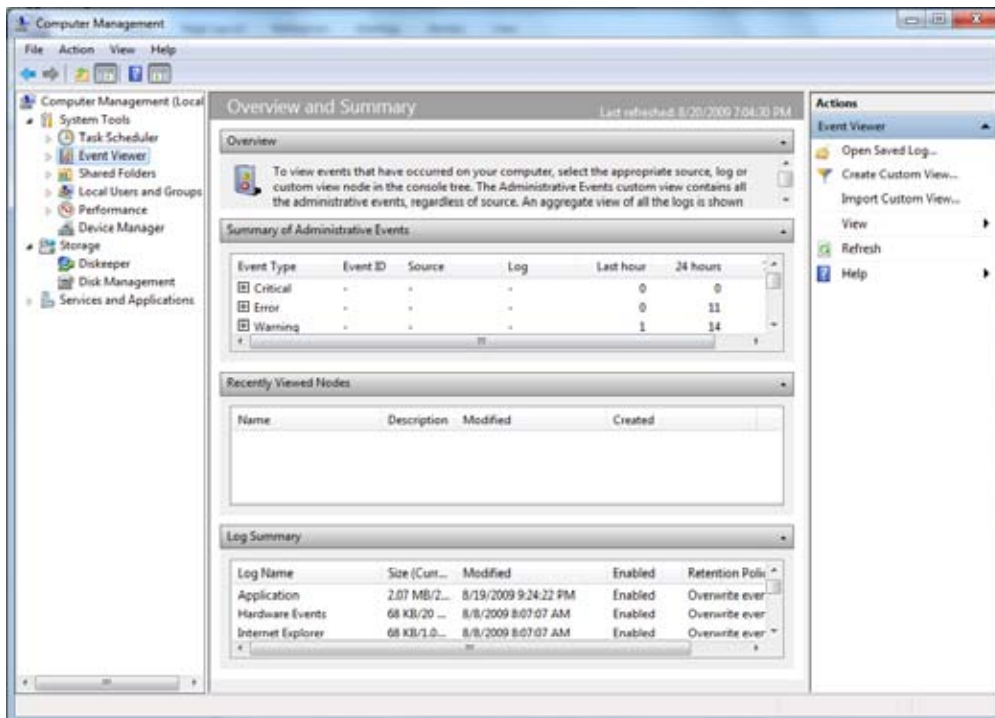
The Computer Management console has a better look and feel in Windows 7. The console is a three-pane interface with navigation, details, and actions, which is similar to what is included in Windows 2008. **Figure E** shows the new Computer Management console.

Figure E



Computer Management console in Windows 7

Figure F



Event Viewer summary

As in previous versions of Windows, the console can operate on the local machine and on remote computers on the same network, making it a one-stop shop full of tools. Most of the tools in the console have been available since Windows 2000, although the tools in Windows 7 are more user-friendly.

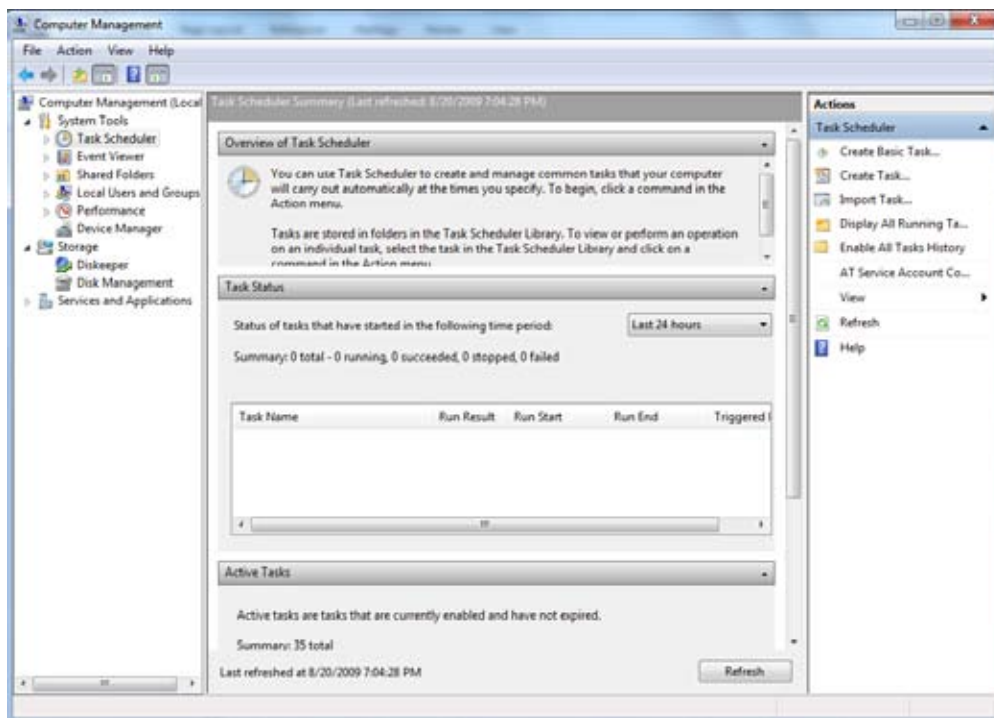
The middle pane of the Computer Management console provides an overview of the option when a utility is first selected and some summary information if it's available, as shown with the Event Viewer selected in **Figure F**.

When moving through the list to use the snap-ins available in the Computer Management console, the Performance Monitor snap-in (listed as Performance in the Console) shows the summary information and details what is happening on the system and directs you to Resource Monitor for further information.

Task Scheduler

Working with Task Scheduler in Windows 7 is a little bit old and a whole lot of new. It is contained in the previously mentioned Computer Management console, but it is quite improved over previous versions of the utility, so I thought I would cover it separately. Task Scheduler, located in the Computer Management Console, features improvements over previous versions of the utility. The view of scheduled tasks, shown in **Figure G**, is somewhat different, and the available actions for tasks are greatly improved.

Figure G



Task Scheduler

The Task Scheduler snap-in can perform the following actions:

- **Create Basic Task:** Uses a wizard to schedule common tasks with few options.
- **Create Task:** Uses a dialog box to specify all details of a task that you are creating. This allows any options needed to be configured.
- **Import Task:** Uses a task created on another computer.

- **Display All Running Tasks:** Shows a summary of running tasks.
- **Enable All Tasks History:** Records the history of all tasks scheduled on the PC.
- **AT Service Account Configuration:** Allows the configuration of the AT service account used by Task Scheduler. The default is the system account.

Note: When creating a task in Windows 7, you can configure it for Windows 7, Windows Server 2008 R2, Windows Vista, or Windows Server 2003, Windows XP, Windows 2000. These configuration options limit the features available in the task to features available within the operating systems selected.

Figure H shows the Create Task dialog box. Many of the items in this dialog box are not new to Windows 7. On the Conditions tab, you can configure conditions for when the task will run; on the Settings tab, you can assign general settings for the task.

The General, Triggers, and Actions tabs are the most changed sections in the Create Task dialog box in Windows 7. I provide more details about those tabs below.

The General tab in the Create Task dialog box is where you name the task and provide a description. You can also specify the task's privilege level (which helps with user account control) and the versions of Windows for which the task is configured.

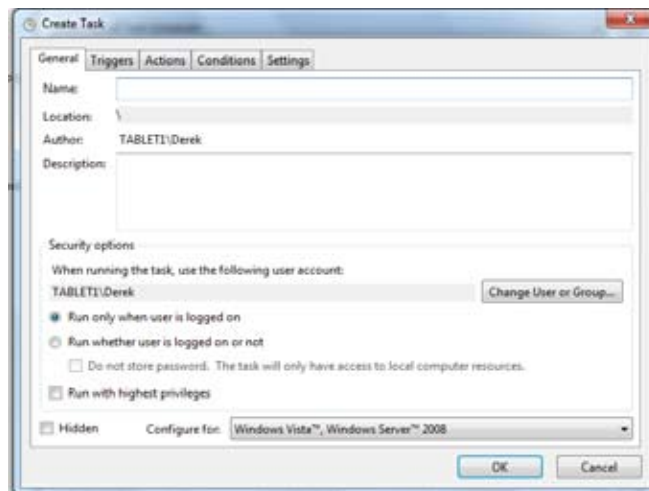
On the Triggers tab, you can specify events that should cause this task to run. This is where the task schedule is defined.

Figure I shows the dialog box for creating a new triggering event for the task.

When deciding how to begin a task or “basing it on a type of event”, Windows 7 has significantly more options than Windows XP. In Windows XP, you could schedule tasks and run them from the right-click menu. In Windows 7, you can schedule a task to happen in any of the following ways:

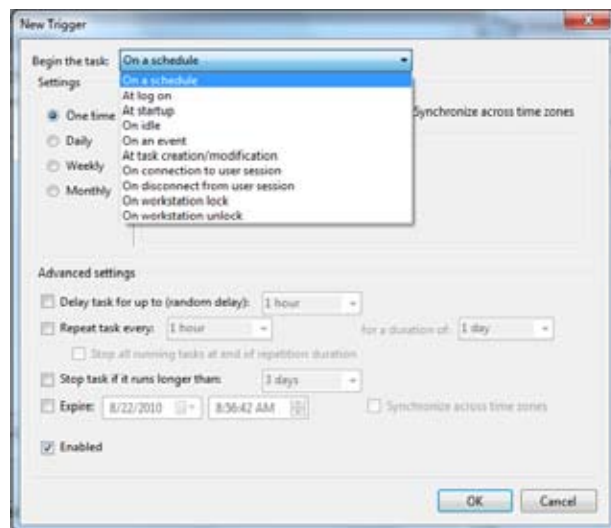
- **On A Schedule:** The task will begin on schedule. (This is the original Task Scheduler setting.)
- **At Log On:** The task will begin when a specific user or any user logs on, based on configuration parameters.
- **At Startup:** The task will begin when the PC starts.

Figure H



Create a scheduled task

Figure I



New Trigger dialog box

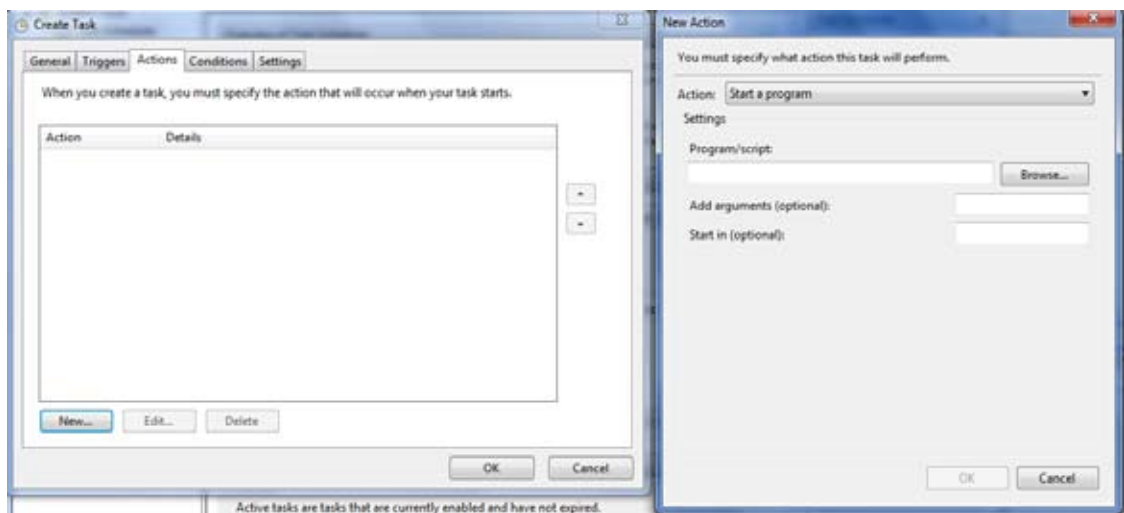
- **On Idle:** The task will begin when Windows deems the computer is idle.
- **On An Event:** The task will begin when a specified event is logged on the system.
- **At Task Creation/Modification:** The task will begin when it is newly added or when it is modified.
- **On Connection To User Session:** The task will begin when a local or remote session is initiated.
- **On Disconnect From User Session:** The task will begin when a local or remote session is disconnected.
- **On Workstation Lock:** The task will begin when a specified user or any user locks the workstation.
- **On Workstation Unlock:** The task will begin when a specified user or any user unlocks the workstation.

The ability to fire a scheduled task on an event being logged is a great feature. Any time a particular machine logs a 1021 error, the administrator can receive an e-mail notification about the issue.

Other settings on the New Trigger dialog box include Advanced Settings, which allow you to specify how long the task should run, and any delay from the time the trigger is fired until the task runs, and the Expiration Date, which allow tasks to be used for a certain period of time and removed. The most important setting in this dialog box is the Enabled checkbox. If the trigger is not enabled, it cannot be used. Fortunately, this box is checked by default.

On the Actions tab, shown in **Figure J** with the New action dialog, you specify what the scheduled task will do when it runs. Your options include the following:

Figure J



The Actions tab of task scheduler and a new action

- **Start A Program:** Starts the specified program (or script). When you choose this option, you need to specify the executable to run. You can also provide a start delay and any required arguments.
- **Send An E-mail:** Allows you to have the task send an e-mail to an administrator or user. You can specify the sender and recipient addresses, subject, message text, any attachments you wish to include, and the SMTP server that will send the message.
- **Display A Message:** Shows the user a pop-up dialog box. When you select this option, you simply provide the message title and text.

Note: A scheduled task can perform more than one action. Click New and complete the New Action dialog box for each action you wish to perform.

When you finish adding your action, click OK.

Windows Memory Diagnostic

The Windows Memory Diagnostic, found in Administrative Tools in the Control Panel, allows you to scan the memory on the system for errors on boot or on a schedule outlined by you. When running the diagnostic utility, the two options, shown in **Figure K**, are to Restart and Check Memory Now or Check Memory Configuration On Next Startup.

BitLocker drive encryption enables you to encrypt and password protect drives used with your system to ensure the security of your data. You can configure the utility to automatically decrypt drives on a specified system, saving you the step of password entry. This feature, which was introduced in Windows Vista remains in Windows 7. However a new feature for portable drives called BitLocker to go has been introduced. The encryption techniques used in BitLocker to go are the same as those in BitLocker, but additional items, such as a viewer are included so portable drives that use the feature can be viewed on non-Windows 7 computers.

Related TechRepublic content: [Secure Your USB Drives with BitLocker to Go for Windows 7](#) and [Windows 7: Mobile Data Protection with Bitlocker To Go](#)

PowerShell 2.0

Microsoft has included PowerShell 2.0 scripting environment in Windows 7, enabling users and administrators to use scripts to manage the computer. Using scripts and commandlets to manage activities in Windows 7 will allow IT Pros to remotely manage the Windows 7 computers in their environment. Many other Microsoft technologies, like Exchange 2010, SQL Server, and even pieces of Active Directory (currently with third party add-ons) can be managed with PowerShell.

PowerShell 2.0 supports an integrated scripting environment, providing a context ready and color coded scripting tool natively. Another great improvement to PowerShell is the remote management capabilities. These include support for fan-in and fan-out remote uses.

Figure K



Windows Memory Diagnostic Options BitLocker

Fan-in Remoting, supports multiple users working against one source server. Think Exchange host with many different administrators hitting their own environment on a shared server.

Fan-out Remoting, supports the issuing of commandlets from one console to multiple servers at the same time. In this case, I have a group of servers I need to run the same script on from one console. I can then fire the script I created against all the servers I need.

Other new features in PowerShell 2.0 include:

- Better WMI command lets **TEXT MISSING?**
- **Synchronous and Asynchronous scripting:** Jobs can be run in the background (or asynchronously) and there is no need to wait for job 1 to complete before running job 2.
- **Script debugging:** Commandlets can be debugged right inside the PowerShell environment making for better coding and less revisions for debugging
- **BYO-Cmdlets:** Build your own command lets. With C# or VB.Net (or some other .Net language) you can create command lets in PowerShell 2.0 with no catch

Related TechRepublic resources: [PowerShell 2.0 — Community Technology Preview \(CTP\) 3 Released](#) and [10 cool things you can do with Windows PowerShell](#)

Windows XP Mode

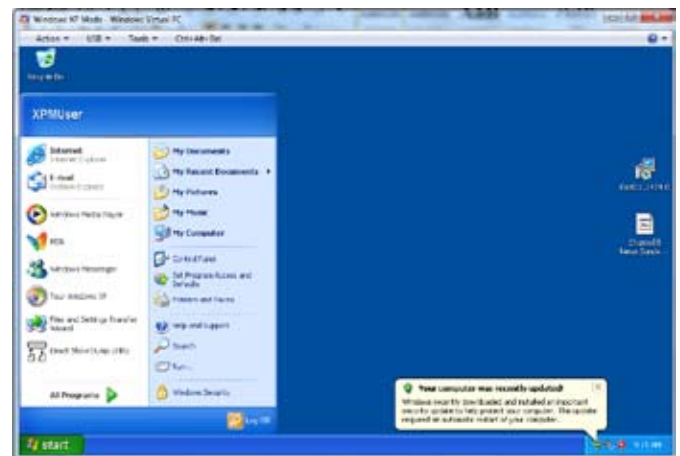
Proposed rewrite: Many IT pros, including myself, still have Windows XP in their organizations because it is more stable and offers better performance than Windows Vista.. Apparently, Microsoft is listening.

Microsoft has introduced a way to keep Windows XP within Windows 7 by using an updated version of Windows Virtual PC and a pre-made virtual hard drive containing a working copy of Windows XP. This was done for compatibility. Microsoft wants to convince organizations that they can migrate to Windows 7 and continue to use all their older applications.

You can seamlessly launch the applications installed in the Windows XP environment from the Windows 7 desktop (or anywhere in Windows 7). The application will launch inside Windows XP, but it will not require your users to work with or even see Windows XP. When an application is installed inside XP Mode, a shortcut to this virtualized application is placed on the Windows 7 Start Menu. When executed, the application will execute inside a VM wrapper and run inside Windows XP Mode. This feature will certainly appeal to those on Windows XP when windows 7 comes out, or those who have applications that weren't ready for Vista.

Figure K shows the Windows XP Mode environment running on my laptop. As you can see, the Windows XP Mode virtual machine is a fully functioning instance of Windows XP.

Figure K



Windows XP will stay with virtual integration in Windows 7.

Figure L shows a virtualized application on the Windows 7 Start Menu. These applications are found under All Programs -> Windows XP Mode -> Windows XP Mode Applications.

When you open a virtualized application while running the XP Mode VM, the application will prompt you to close the Virtual Machine to continue. Clicking OK closes the VM and starts the application.

Note: The Windows XP Mode virtual machine stores the logon credentials created the first time you configure it, making it easy to access and handle the logon for the user.

Related TechRepublic resources: [Determine if your hardware can support Windows XP Mode in Windows 7](#) and [10 reasons why Windows 7's XP Mode is a big deal](#)

Windows installation from a USB flash drive

I'm the only IT professional in my organization, which makes for quite a challenge where installations and deployments are concerned. Being able to carry a flash drive containing the installation media of an OS is much more portable and convenient than CD or DVD media in many cases.

Using utilities such as DiskPart and the Windows 7 installation media (or image), you can create a bootable flash drive to provide a small, ultra portable set of media for installing Windows 7 on machines that may not have access to all the resources within your environment.

Related TechRepublic resource: [Configure a USB drive to be a Windows 7 installation platform.](#)

Boot Windows 7 from a VHD file

Windows 7 supports booting from a Virtual Hard Disk (VHD) file, which can speed up deployments and configurations. You can create VHD files containing Windows 7 and all the applications needed for a specific PC. Then these VHD files can be copied to laptops or desktops throughout the environment and to the machines configured to boot from the VHD by adding these installations to the Boot Manager.

Related TechRepublic resource: [Windows 7 now allows you to boot a native OS from a VHD file](#)

Summary

The list of features and tools provides you with a good starting point and reference for your Windows 7 migration either in testing or in production.

Figure L



Seamlessly virtualized applications in XP Mode on the Start Menu

Explore the Problem Steps Recorder in Windows 7

If you've ever worked a help desk and become extremely frustrated while trying to coax an end user into accurately describing the problem that has been encountered, you are going to love a new tool in Microsoft Windows 7 called the Problem Steps Recorder. When started, this new tool will essentially record each and every step a user takes and document the entire operation in both screen captures and step-by-step details. When stopped, the Problem Steps Recorder will save the recorded information as a compiled HTML file and package it up in a ZIP file that the end user can then e-mail to the help desk.

Here's an introduction to Windows 7's Problem Steps Recorder.

Launching the tool

At this point in the beta, finding the Problem Steps Recorder is a bit tricky because it really does not have a prominent access point. That may change as the UI matures a bit, but then again, it may remain hidden until needed — such as when a help desk tech tells the user to run it.

In any case, you can launch the Problem Steps Recorder in the beta by typing PSR in the Start menu's Search box and pressing [Enter]. However, I also located it in the Control Panel by searching with the keyword "Problem," as shown in **Figure A**.

Recording a problem

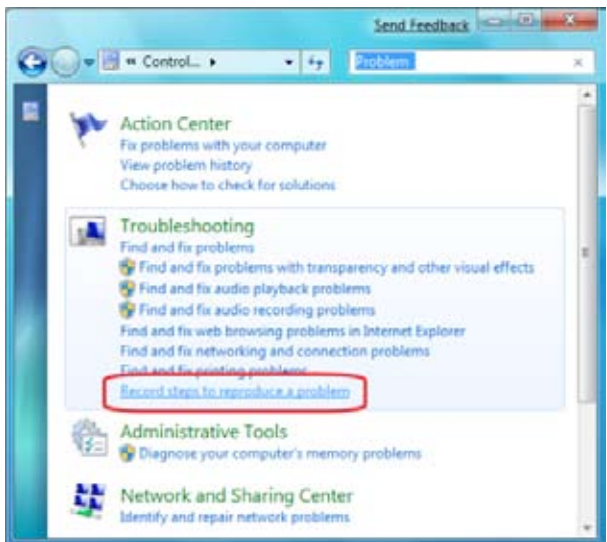
The user interface for the Problem Steps Recorder in the beta is very straightforward and bears a close resemblance to Sound Recorder, as shown in **Figure B**. To begin a recording operation, you just click the Start Record button.

Once the Problem Steps Recorder has begun recording, its title bar flashes the words Recording Now and the icon on the Taskbar shows a red flashing dot. At this point, you can begin carrying out the steps that lead to the problem.

To experiment with Problem Steps Recorder, I'm going to simulate an error condition by setting the screen orientation to Portrait (flipped), which is a setting that my monitor doesn't support.

As you can see in **Figure C**, I right-clicked on the desktop and launched the new Screen Resolution tool. If you shift your attention to the Problem Steps

Figure A



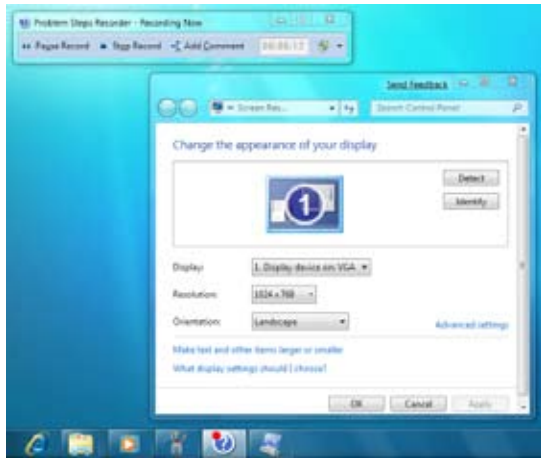
Clicking the Record Steps to Reproduce a Problem link launches the Problem Steps Recorder.

Figure B



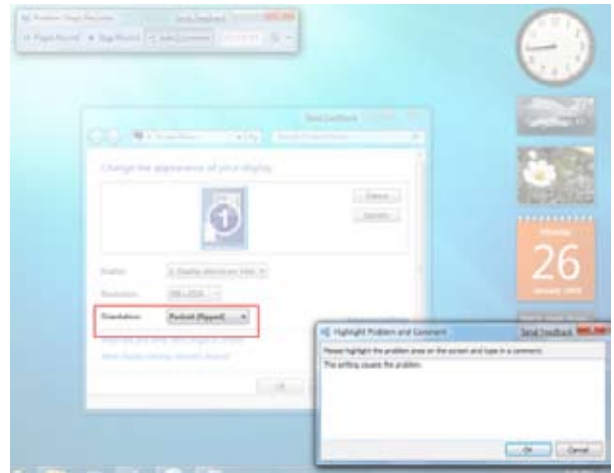
The Problem Steps Recorder's user interface is very straightforward.

Figure C



Once you start a recording operation, the buttons on the Problem Steps Recorder change.

Figure D



The Add Comment feature will make it easier for users to annotate the problem area if additional information is required.

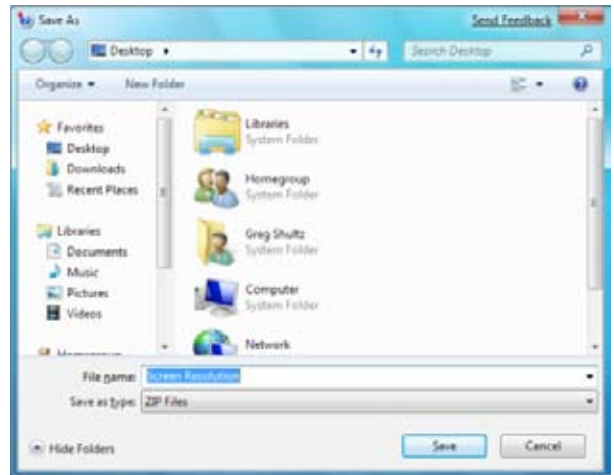
Recorder, you'll notice that the Start Record button has been replaced by the Pause Record button, the Stop Record and Add Comment buttons are now active, and the time counter has begun counting.

Right next to the counter there is a UAC shield icon. Clicking this icon will, of course, yield a UAC, which, once responded to, will allow the Problem Steps Recorder to run as an administrator and make it possible to record interactions with programs that are running in administrator mode. In other words, the Problem Steps Recorder will be able to keep running in the background while you respond to the UAC.

Getting back to my example, as I began working through the steps of changing the orientation, I clicked the Add Comment button, which allowed me to highlight a particular area of the screen and annotate it with a command, as shown in **Figure D**.

Once I completed my test and closed the Screen Resolution tool, I clicked the Stop Record button. At that point, the Problem Steps Recorder displayed the Save As dialog box, shown in **Figure E**, and prompted me to name the file, which it then saved as a Zip file.

Figure E



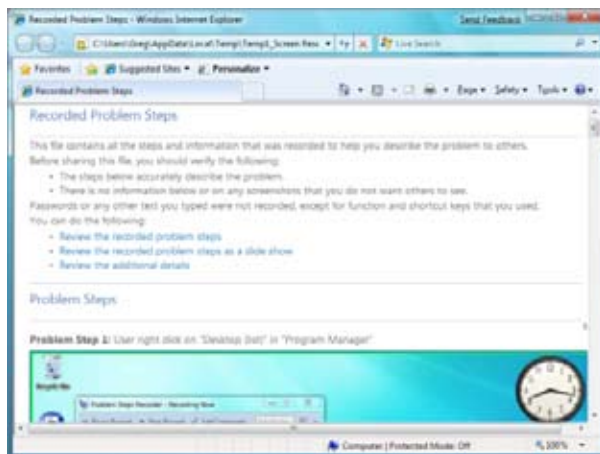
The recorded session is automatically saved in a Zip file.

Figure F



When you open the Zip file, you can see and launch the compiled HTML file.

Figure G



You can then view the entire recorded session with Internet Explorer.

When you open the Zip file, you'll see the compiled HTML file, as shown in **Figure F**.

When you double-click the compiled HTML file, Internet Explorer will launch and open the recorded session and display the screen shots and a step-by-step account of the entire procedure. **Figure G** shows the beginning of the file, but you can download the Zip file and view the entire session in the accompanying TechRepublic download.

Take advantage of tools to monitor memory usage in Windows 7

Ever since I discovered how to use the Windows 9x's Resource Meter to keep track of system resources and prevent system crashes caused by depleting system resources when running more than one application at a time, I've made good use of Windows monitoring tools.

Over the years Microsoft has vastly improved the primitive Resource Meter and increasingly made it easier to see what is going on with the operating system's use of memory. With Windows 7, I've discovered some slight, but notable, changes in the way that memory usage is reported on the Performance tab of the Task Manager. I have also noticed a very nice chart in the new Resource Monitor that provides very detailed information on physical memory usage at a glance.

Let's take a closer look at the changes on Task Manager's Performance tab and investigate the Physical Memory usage chart.

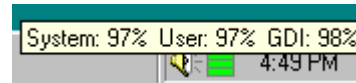
Taking a look back

To help you to appreciate the memory usage information in Windows 7 and to put the evolution of Windows resource monitoring into perspective, I thought I would begin by taking a quick look back at Windows 9x's Resource Meter. Once you have Resource Meter up and running, you can hover the mouse pointer over the icon and you will see a pop-up display that breaks down the amount of available system resources into its three categories: System, User, and GDI, as shown in **Figure A**.

If you double-click the icon, you will see the Resource Meter dialog box, shown in **Figure B**. As you can see, this display used three gas gauge-like monitors to show you the amount of available system resources in each category.

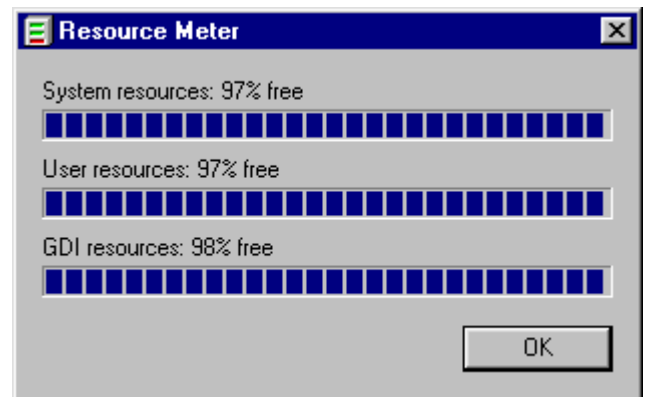
Because resource management was horrible in the Windows 9x days, system crashes were inevitable. If you kept your eye on the Resource Meter icon, you could literally watch the amount of system resources depleted as you worked. As you can see in **Figure C**, the green bars would move down and change to yellow and then to red right before the system crashed.

Figure A



Hovering the mouse pointer over the Resource Meter icon produces a display of the currently available system resources in three categories.

Figure B



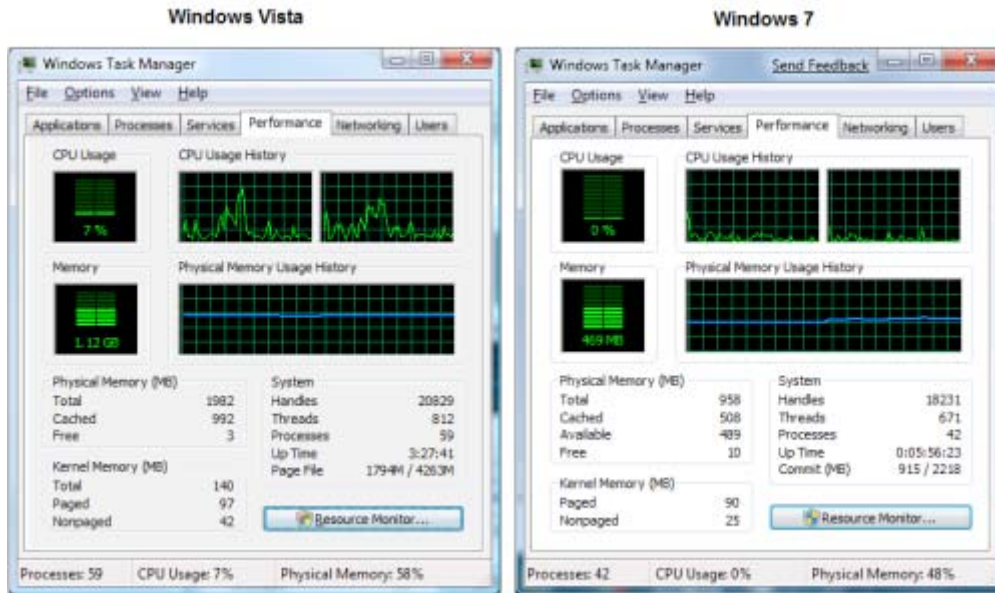
The Resource Meter's display uses three gas gauge-like monitors to show you the amount of available system resources in each category.

Figure C



This is the evolution of a system crash caused by the depletion of system resources.

Figure D



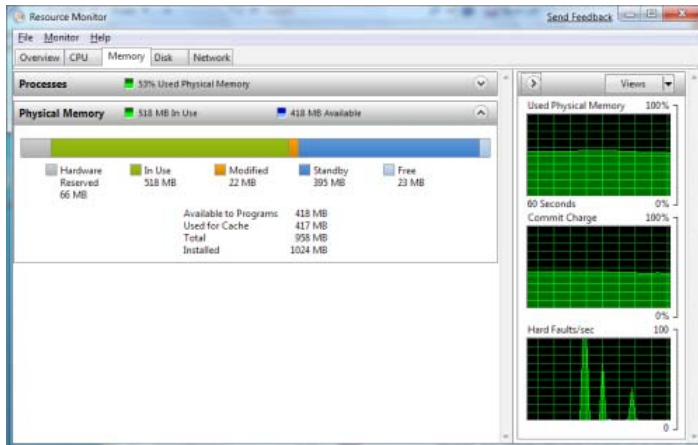
Comparing Vista's and 7's Performance tab reveals a few slight, but noteworthy, changes.

The Task Manager Performance tab

When you compare the Performance tab in Windows 7's Task Manager to the one in Vista, as shown in **Figure D**, you'll notice a few changes in the statistics sections. First, in the Physical Memory section, you'll see that there is now a measurement showing the amount of available memory. While in Vista you can easily deduce the amount of available memory by subtracting the Total value from the Cached value. It is nice to be able to tell at a glance just how much physical memory is currently available to the system.

Of course, the Total entry shows the amount of RAM installed in the system. The Cached entry indicates the amount of physical memory used recently for system resources. (This memory will remain in the cache in case the system resources are needed again, but it's available should other operations need it.) The new Available entry indicates the amount of physical memory that is currently not being used. The Free entry indicates the amount of memory being used in the cache that does not contain useful information.

Figure E



The Physical Memory usage chart provides a very easy to understand picture of memory usage.

The Kernel Memory section now shows only the amount of Paged and Nonpaged memory. The Paged and Nonpaged entries break down the total amount of memory being used by the kernel and show you how much is coming from virtual memory and how much is coming from physical memory, respectively. The Total entry, which was essentially useless here, has been removed.

In the System section, you'll find the same measurements found in Vista; however, the Page File item has been renamed to Commit. (Windows XP called this same measurement the Commit Charge and dedicated an entire section to it.) This measurement still shows Page File usage. Here, the first number indicates the total amount of physical and virtual memory currently in use, while the second number indicates the total amount of physical and virtual memory available on your computer.

The Handles and Threads entries are rather obscure and are actually sub-objects of processes. The Handles entry shows the number of object identifiers, or handles, that are currently in use by all the running processes. The Threads entry actually refers to the number of subprocesses running inside larger processes. The Processes entry, of course, represents the number of currently running processes. As you know, you can see each of the currently running processes by selecting the Processes tab.

The Up Time entry is still a nice piece of information that shows the amount of time that has passed since the computer has been restarted.

The Resource Monitor Memory tab

To access the Resource Monitor, just click the button on the Performance tab in Windows 7's Task Manager. Then, select the Memory tab in the Resource Monitor. When you do, you'll see the newly revamped interface, shown in **Figure E**. While the three graphs on the side are a nice new feature and show regularly updated graphs, the Physical Memory usage chart is stunning in that it provides an extremely intuitive picture of memory usage.

If you do the math, you can see the chart accounts for every bit of memory installed in the system. **Table A** identifies each section of the chart. If you subtract the amount reserved for hardware, you get the Total. If you add the Standby and Free, you get the Available to Programs total.

Table A

Hardware Reserved	Memory that is reserved for use by the BIOS and some drivers for other peripherals.
In Use	Memory used by process working sets, drivers, nonpaged pools, and operating system functions.
Modified	Memory whose contents must be to disk before it can be used for another purpose.
Standby	Memory that contains cached data and code that is not actively in use.
Free	Memory that does not contain any valuable data and that will be used first when processes or the operating system needs more memory.

Secure your USB drives with BitLocker To Go for Windows 7

When Microsoft introduced Windows Vista, one of the big security features in that operating system was BitLocker, a hard drive encryption scheme designed to protect sensitive data from being accessed on lost or stolen computers — mainly laptops.

With the huge increase in the use of very small, large capacity, USB drives, the potential for sensitive data to be lost or stolen has really become more of a problem because it is much easier to lose or steal a device no bigger than a package of chewing gum. To protect sensitive data stored on USB drives, Microsoft Windows 7 features the encryption scheme called BitLocker To Go.

In this article, I'll introduce you to BitLocker To Go and show you how it works on a 1GB USB thumb drive.

How it works

Basically, BitLocker To Go allows you to encrypt a USB drive and restrict access with a password. Without the password, the USB drive is worthless. When you connect the USB drive to a Windows 7 computer, you are prompted for the password and upon entering it you can read and write to the drive as you normally would.

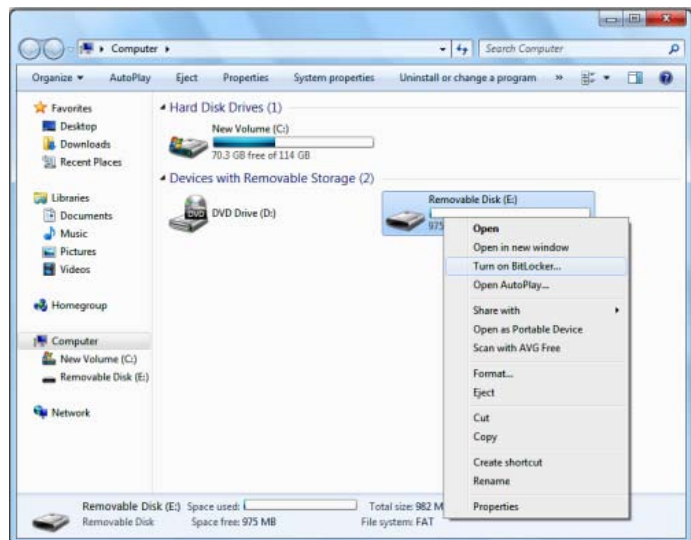
During the encryption process, Windows 7 installs a special reader on the USB drive. When you connect the USB drive to a computer running XP or Vista, the BitLocker To Go Reader takes control, prompts for the password, and then basically makes the USB drive a read-only device.

BitLocker To Go can be used by both home and business users. In a Domain system, IT administrators can configure a policy that requires users to apply BitLocker protection to removable drives before being able to write to them. Furthermore, the policy can specify password length as well as complexity.

Setting up a USB drive

Setting up BitLocker To Go on a USB drive is a simple procedure. Once you insert a USB drive, right-click on it and select the Turn on BitLocker command from the menu, as shown in **Figure A**.

Figure A



When you right-click on a USB drive in Windows 7, you'll see the Turn on BitLocker command.

As soon as you do, BitLocker To Go will begin initializing your USB drive. The process is nondestructive, so you don't have to worry about any data that is already on the drive.

Once the initialization process is complete, BitLocker To Go will prompt you to set up a password that you will use to unlock the drive. If you have a smart card, you can use its PIN to unlock the drive.

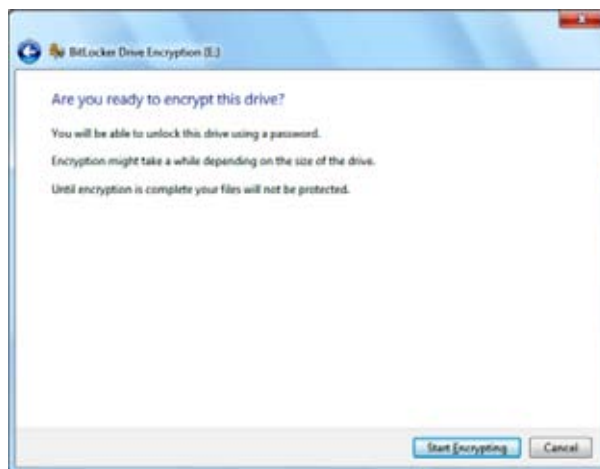
After you set up a password or use a smart card, BitLocker To Go will prompt you to store a recovery key. You can use the recovery key to unlock your drive in the event that you forget the password or lose your smart card.

When you create the password and save your recovery key, you'll be prompted to begin the encryption process, as shown in **Figure B**.

During the encryption process, you'll see a standard progress monitor that will keep you apprised of the operation. The amount of time that it will take to complete the process will depend on how large the drive is. As you can see, there is a Pause button that will allow you to temporarily halt the process should you need to perform another task.

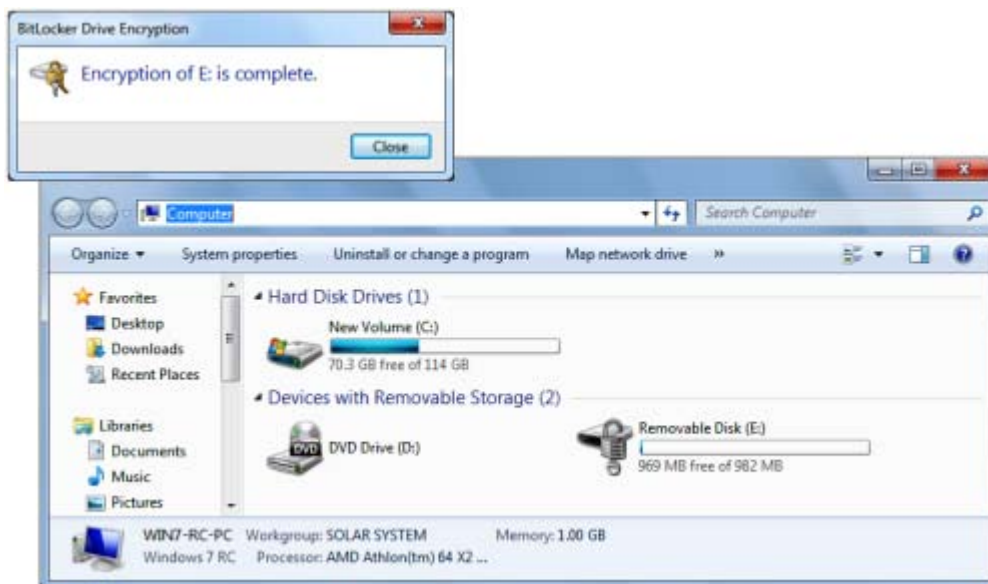
Of course, once the encryption is complete, BitLocker To Go displays a confirmation dialog box and changes the icon associated with the encrypted drive, as shown in **Figure C**.

Figure B



You'll be prompted to begin the encryption process once you save the recovery key.

Figure C



When the encryption is complete, you'll notice that the drive icon shows a lock on the drive.

Using a BitLocker To Go encrypted drive in Windows 7

When you later insert the BitLocker To Go encrypted drive in the Windows 7 system, you will immediately be prompted to enter the password, as shown in **Figure C**. If you wish, you can select the Show Password Characters as I Type Them check box, so that you can see the letters; otherwise, you'll see asterisks. After you type the password, you can select the Automatically Unlock on This Computer from Now On check box to store the password in Windows 7's password cache.

Once you click Unlock, you'll see an AutoPlay dialog box that prompts you to view the files or use ReadyBoost. When you click the Open Folder to View Files button, you will be able to access the drive and its contents as you normally would.

Using a BitLocker To Go encrypted drive in Windows XP/Vista

When you insert the BitLocker To Go encrypted drive in a Windows XP or Vista system, you will see an AutoPlay dialog box that prompts you to install the BitLocker To Go Reader. When you click this button, it will take just a moment to install and run the Reader.

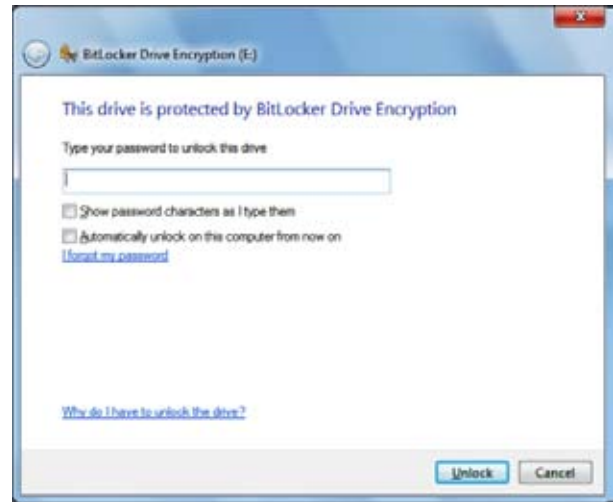
You'll then see the BitLocker To Go Reader dialog box, which will prompt you to enter your password. Notice that the Automatically Unlock on This Computer from Now On check box is missing from this dialog box. However, the Show Password Characters check box is still available.

After you type the password and click the Unlock button, you'll see the BitLocker To Go Reader window, which essentially looks like Windows Explorer. However, it doesn't work like Windows Explorer.

If you attempt to open any file by double-clicking it in the BitLocker To Go Reader window, you'll immediately be prompted to copy the file to the desktop — you won't be able to open the file on the USB drive.

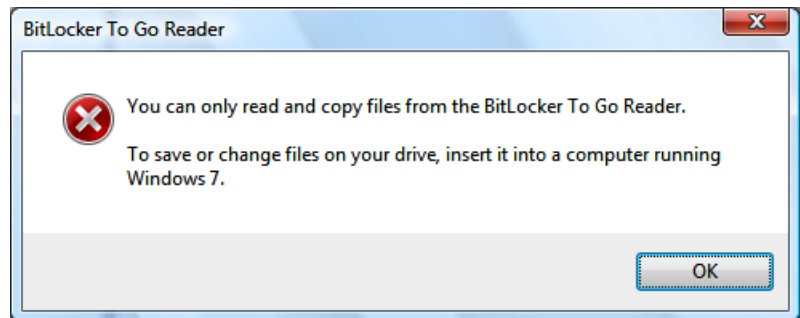
If you attempt to copy a file from the computer to the BitLocker To Go Reader window, you'll immediately see the error message shown in **Figure D**.

Figure C



When you insert the BitLocker To Go encrypted drive in Windows 7 system, you will immediately be prompted for a password.

Figure D



You cannot copy files to an encrypted drive from the BitLocker To Go Reader.

Windows 7: Mobile Data Protection with Bitlocker To Go

Windows 7 contains several new features, including one I'm pretty excited about: Bitlocker To Go. Expanding the Bitlocker offering in Vista, Microsoft provides encryption support for removable USB storage—a wide channel for escaping data—on the Enterprise and Ultimate versions of its new, still in beta, OS.

Bitlocker ToGo Features Overview

Bitlocker To Go (BtG) provides a right-click selection to enable encryption, as shown in **Figure A**. The drive selected is a Lexar 512MB USB flash drive.

Encryption time is pretty fast, and the final result is a mobile storage device which can be accessed only with a password, passphrase, or PIN. And access isn't limited to Windows 7 systems.

Files on a BtG protected device are not only accessible by Windows 7 systems. They are also available using Windows XP and Vista. Files written to the drive during the encryption (shown in **Figure B**) process make this possible. If auto-running applications when connecting a USB drive is disabled on your computer—and it very well should be—selecting the drive via Windows Explore displays the BtG application (arrow). The files shown in this picture are only items accessible until the drive is unlocked.

Unlocking a BtG encrypted device on a Windows 7 system allows a user to see it as standard drive, with both read and write privileges. However, using XP or Vista provides read only access, requiring copying a file to the local drive before access is granted.

No action is necessary to relock a drive other than removing it from the USB port. Security managers don't have to worry about someone forgetting to lock a device or copy a file to a secure area before taking it on the road. Further, use of BtG on any connected USB storage is controllable with group policy object settings.

So what didn't I like? Not much, actually. The biggest issue I had during testing was the lack of an un-encrypt function to restore the drive to an unprotected state. It might exist, but like other

Figure A

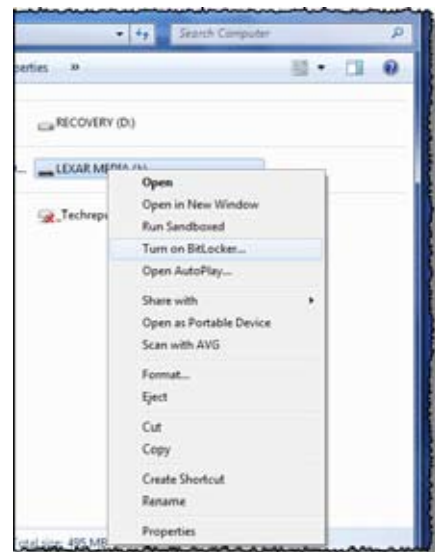


Figure B



Figure C



functions in Windows 7 beta it might simply be in an inconvenient (where I can easily find it) location. Otherwise, I think this is a great new feature; one I wish was available on all versions of Windows 7 and Vista.

If you're interested in a more detailed description of how to use BtG, read on.

Setup and testing

For my test, I used a Dell desktop running Windows 7 beta (Ultimate) and a Dell laptop running Windows XP SP2. There's nothing to download or set up to get started. BtG is part of the standard install.

I plugged the 512MB Lexar drive into a USB port on the desktop and waited for Win7 to set it up for use. I then brought up the drive list, right-clicked on the Lexar, and selected Turn on Bitlocker, as shown in Figure A.

The first window to appear is shown in **Figure C**. I had a choice between using a password, a smart card, or both. I chose password only. The application checks the strength of the password entered. If it doesn't meet the requirements for a strong password, you can't proceed until one is entered which does. If in doubt, help is available in Win7 which describes what makes up an acceptable password or passphrase, as shown in **Figure D**.

After entering a strong password, the drive was encrypted (less than two minutes) and I was asked how I wanted to store my unlock code. The unlock code is used when the password or passphrase is forgotten or a smart card lost. As shown in **Figure E**, I could save it to a file or print it. Although Microsoft recommends doing both, I opted to only store it in a text file on my desktop's local drive. And that's it.

Figure D

What makes a password or passphrase strong?

A strong password:	A strong passphrase:
<ul style="list-style-type: none"> • Is at least eight characters long. • Does not contain your user name, real name, or company name. • Does not contain a complete word. • Is significantly different from previous passwords. 	<ul style="list-style-type: none"> • Is 20 to 30 characters long. • Is a series of words that create a phrase. • Does not contain common phrases found in literature or music. • Does not contain words found in the dictionary. • Does not contain your user name, real name, or company name. • Is significantly different from previous passwords or passphrases.

Strong passwords and passphrases contain characters from each of the following four categories:

Character category	Examples
Uppercase letters	A, B, C
Lowercase letters	a, b, c
Numbers	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Symbols found on the keyboard (all keyboard characters not defined as letters or numerals) and spaces	^ ~ ! @ # \$ % ^ & * () _ - + = { } [] \ : ; ' ' < > , . ? /

Figure E



Figure F



To test unlocking the drive, I removed it from the USB port, waited a second or two, and plugged it back in. Win7 immediately saw it as a BtG enabled device, and displayed the unlock window shown in **Figure F**.

The first time I tried this, I entered my password and the drive was instantly ready for use. During the second test, I clicked I forgot my password which brought up the unlock key entry screen in **Figure G**.

Figure G



Note the recovery key identification code. This is written to the drive when encrypted to identify the device. It is also written to the text file or printed with the recovery key so you know which recovery key goes with which drive. I copied and pasted the recovery key from the text file created earlier. The next step provided the option of changing the password or authentication method to facilitate future access.

Figure H



Since this worked flawlessly, it was time to test accessing the drive from my XP-based laptop. Since autorun is disabled, I had to select the drive from Windows Explorer and manually run BtG, as shown in Figure B. This brought up the unlock window in **Figure H**.

Note the difference between the XP unlock and the Win7 unlock shown in Figure F. At any time I can tell BtG on Win7 to automatically unlock my device. This is not an option for XP.

I entered my password, and an Explorer-like window appeared telling me to copy the files I wanted to view to the local drive. This is necessary on non-Win7 systems or the files are not accessible. I also attempted to copy a file back to the drive without success (not a supported feature, but you never know...).

Overall, I found BtG a great addition to Windows for both individual and business users.

PowerShell 2.0 - Community Technology Preview (CTP) 3 released

Although version 2.0 of Microsoft's PowerShell scripting language has not yet made it to the beta stage, Microsoft has released a third community technology preview of the shell. Designed to provide those interested a sneak peak as to what might make its way into the final release of version 2, it's important to note that a community technology preview is not a beta and features may come and go. Microsoft makes it very clear that CTP software is not in any way neither designed nor supported for production use and that CTP software may not even closely resemble final code. In short, don't build solutions on CTP software and then complain when it breaks later on!

What's coming in PowerShell 2.0?

PowerShell 2.0 is packed full of new features designed to make it a more powerful, easier to use scripting language than the initial 1.0 release. I'll discuss a few of the more significant enhancements here.

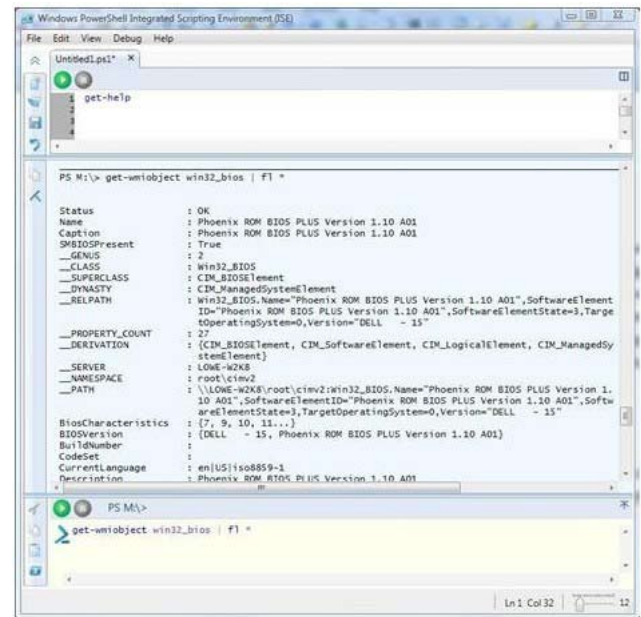
ISE

The most immediately noticeable addition to PowerShell 2.0 comes in the form of the Windows PowerShell Integrated Scripting Environment (ISE). ISE is a GUI-based PowerShell script development tool designed to make it much easier to create and modify PowerShell scripts. Although there are other free third party development solutions, such as Quest's PowerGUI, it's nice to see the PowerShell team working to tame the beast a bit.

Remoting

PowerShell 2.0 will also greatly enhance the ability for a PowerShell script to work against remote systems. In short, an administrator can, from his own workstation, issue PowerShell commands that are executed on remote stations, with the output being returned to the administrator's own screen. Of course, in order for remoting to work, the remote computer must have PowerShell installed. The beauty of remoting in PowerShell 2.0 is that the remote user doesn't see any visual representation that a PowerShell process is being executed, beyond the actual PowerShell process appearing in Task

Figure A



PowerShell 2.0 ISE

Manager. If user's did see screens start popping up on their desktops in the middle of a work day, imagine what would happen at the help desk when the phone started ringing!

Background jobs

If you take a deeper look at the remoting functionality in PowerShell 2.0, you'll understand that this new feature works because of PowerShell 2.0's ability to run jobs in the background, shielded from the user. Locally (or remotely, for that matter), administrators can create scripts that run in the background; that is, the script executes and the administrator is immediately returned to a new PowerShell prompt while processing continues in the background. Later, when time permits, the administrator can gather the results of the command for further action. The ability to run jobs in the background — or asynchronously — means that an administrator can keep working without constantly waiting for scripts to finish execution.

Transactions

PowerShell 2.0 also adds the ability to develop full transaction-based scripts complete with cmdlets for starting, committing, and rolling back a transaction in the event of an error or other condition. This capability starts to make PowerShell an attractive choice for even the most complex business processes as administrators can be more sure that scripts can return data to its original state if necessary.

Multiline comments

In PowerShell V1, comments that spanned lines were, quite frankly, a pain in the neck unless you used a tool like PowerGUI. Each comment line had to start with a # symbol. Failure to include one of the symbols meant a bad script. PowerShell 2.0 introduces the <# and #> script elements between which PowerShell script developers can embed as many comments across as many lines as they wish. While it doesn't sound like a major improvement, if you look at almost any serious programming or scripting language, they all have the ability to handle multiline comments.

New cmdlets, parameters, variables and operators

PowerShell 2.0 is the next major release of the language and, as such, includes a huge number of new cmdlets, new parameters for existing cmdlets, new permanent variables and some new operators, such as the -Split and -Join operators that allow the splitting or joining of strings based on certain characteristics. There are new cmdlets to support background jobs (Start-PSJob, Stop-PSJob, etc.), to support transactions (Start-PSTransaction, Undo-PSTransaction, etc.), and new cmdlets to support PowerShell 2.0's enhanced debugging capabilities (Set-PSBreakpoint, Get-PSBreakpoint, etc.) As I said, some existing cmdlets have also received new parameters. For example, the Stop-Process cmdlet now has a -force parameter. The Export-Csv cmdlet now sports a -Delimiter parameter so you can specify what character you want to use. While many of the new cmdlets and parameters are designed to support the new overall functionality in PowerShell, many enhancements also make PowerShell 2.0 a much more robust language overall.

Summary

PowerShell 2.0 has been in development for a long time and, with the third CTP under its belt, it's looking like a huge step in the right direction for this excellent scripting language.

10 cool things you can do with Windows PowerShell

If PowerShell's learning curve has kept you from embracing it for daily use, "cool" might not be a word you'd associate with it. But PowerShell is here to stay. It's a core part of Exchange 2007, Windows Server 2008, and SQL Server 2008, and it has immense power we all need to grasp.

I'm going to put some fun into the PowerShell arena and show you a few tricks that will definitely come in handy. Besides, it is always cooler when you amaze someone with the solution from the command line. Having someone watch you right-click and fix something doesn't have the same appeal.

Note: Be careful, very careful

Yes, this is a tool worthy of the name. PowerShell can easily cause massive configuration changes, positive or negative — so protect yourself and establish a test environment for your learning experiences. Also consider using the "-confirm" parameter to test configurations before execution for certain commands.

1. Report all of the USB devices installed

PowerShell is Windows Management Instrumentation (WMI) aware. From PowerShell, you can make a WMI call to retrieve the USB devices installed in a local or remote system:

```
gwmi Win32_USBControllerDevice -computername SERVER1 |fl  
Antecedent,Dependent
```

This will apply a filter to bring back the antecedent and dependent fields from the SERVER1 computer. Should you want the full export, you can omit the pipe and filter statement to have a comprehensive export of the USB devices on a system. I have found this useful to maintain a report for servers that have a USB license device installed so that their connectivity is maintained from the device perspective.

2. Perform your favorite CMD tasks in PowerShell

Yes, you can stop using the DOS prompt and start doing all of those same tasks within PowerShell. This can make learning a little easier and help you become more familiar with the interface. Unfortunately, from the run prompt, there is no three-letter launcher like *cmd*. But *powershell* will launch it. You can also assign a shortcut key to PowerShell so Ctrl + Shift + P launches it directly.

3. Kill a process in PowerShell instead of Task Manager

When you have a Windows service running that will not respond to stop commands, you can use PowerShell to perform the equivalent actions of ending the task within Task Manager. For instance, you'd do the following for BadThread.exe:

```
get-process BadTh*
```

The results will be similar to this:

Handles	NPM (K)	PM (K)	WS (K)	VM (M)	CPU (s)	Id	ProcessName
28	4	-210844	-201128	-163	25.67	2792	BadThread

Once the Process ID has been identified, you can kill the errant process by entering:

```
stop-process -id 2792
```

At that point, the BadThread example will be hard stopped and you can resume your attempt to start the service. You can do that right here in PowerShell as well.

4. Use PSDrive to view more than just drives

The PSDrive command lets you view objects of the Windows environment beyond traditional network, local, or removable drives. One popular view is the HKLM PSDrive to view the HKEY_LOCAL_MACHINE top-level hive of the registry. To get into the registry, enter the following command:

```
PS C:> cd HKLM:  
PS HKLM:/>
```

You are then transported into the registry hive and can view and even delete items, should you wish.

5. Export NTFS folder permissions — recursive or not

Managing NTFS permissions is a whole separate matter, but with PowerShell, you can export the permissions to audit access or take a quick look at access control lists (ACLs) for the security configuration. This can be a great accountability mechanism to run in a scripted format periodically — or you can run it on demand to diagnose a particular issue. For example, take the following iteration:

```
PS E:>Get-Acl N:Data
```

This will give you a quick report of your security rights to the specified path (note that it won't give the share access). That alone is nothing too exciting, as it will report only the single specified path, but if you want to include recursion for the entire path, you can use other strategies. For the same path (N:\Data), you'd use the Get-ChildItem command (cmdlet) within PowerShell, combined with the Get-Acl command. Consider the following example:

```
PS E:>Get-ChildItem N:Data -recurse | Get-Acl
```

This will span the entire N:\Data path and display the ACLs for the contents of the path. What happens here is that the Get-ChildItem provides an inventory of the file system objects, and that collection is passed to Get-Acl to provide the results for each item.

If you want to archive this to a comma-separated variable (CSV) document, you pass “| export-csv c:\filename.csv” at the end of the cmdlet. You can also pass the normal “> C:\filename.txt” to the end of the command to get it exported to a text file. Note that when you use the -recurse option, it does just that and will traverse the entire path you specify. So be careful when doing it across a large volume or over the network.

6. Play with PowerShell 2.0

PowerShell 2.0 is in the Community Technology Preview (CTP) stage. It includes a graphical interface, Graphical PowerShell, and it is cool. The PowerShell scripts are saved as .ps1 files, making it easy to modify, import, and transfer scripts across systems. **Figure A** shows our NTFS permissions example while running in the graphical mode.

One note on PowerShell 2.0: You have to configure the execution policy through PowerShell (nongraphical version) before using the tool. Configure one of the following execution policies:

```
PS C:>Set-ExecutionPolicy Restricted  
(check only)
```

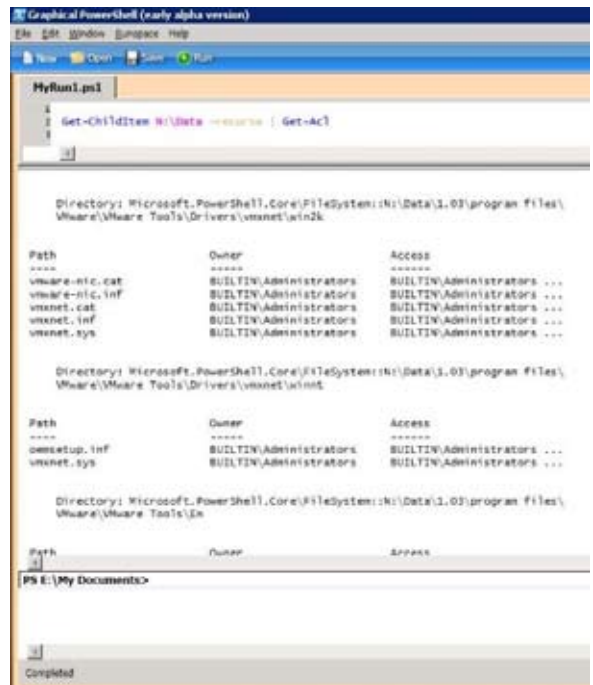
```
PS C:>Set-ExecutionPolicy AllSigned  
(most secure)
```

```
PS C:>Set-ExecutionPolicy RemoteSigned  
(medium secure)
```

```
PS C:>Set-ExecutionPolicy Unrestricted  
(least secure)
```

When deciding to evaluate PowerShell 2.0, note that the WS-MAN v1.1 package is required, and if you want to use the graphical interface, Microsoft .NET Framework 3.0 is required.

Figure A



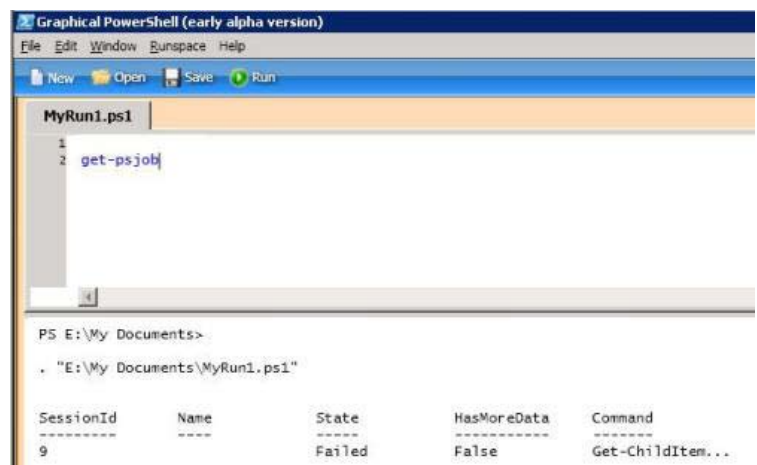
7. Work from the keyboard in Graphical PowerShell

If you are familiar with the Microsoft SQL Query Analyzer environment, you will appreciate some of these keyboard shortcuts. In Graphical PowerShell, you can select a single line or multiple lines and execute them by pressing the F5 key. Also, if you have modified your script, the familiar Ctrl + S to save, Ctrl + Z to undo, Ctrl + C to copy, and Ctrl + V to paste are available to save you time in the editing and testing.

8. Background a time-consuming task

If you have a cmdlet that will take some time to run, you can use PowerShell to send it to the background to complete. In this way, you can send a series of commands to execute at once and let them complete on their own schedule. The command to launch

Figure B



a background job leads with the `start-psjob -command` parameter. You can query PowerShell on the status of any of the jobs with the following command:

```
PS C:>get-psjob
```

You'll see a table of results showing the current status of your jobs, with a session identifier that is unique for each job. **Figure B** shows one failed job.

You can remove the failed job by running the following command:

```
PS C:>remove-psjob 9
```

9. Insert timestamps into PowerShell outputs

For your PowerShell tasks, you can have a timestamp entered in series so you can determine how long a single step occurs or to use as a logging mechanism for your scripts. I find this handy in Graphical PowerShell when I'm testing scripts. To insert a timestamp, enter one of the following commands as a single line within your .ps1 file:

Command	Output example
<code>"\$(Get-Date -format g) Start logging"</code>	2/5/2008 9:15 PM
<code>"\$(Get-Date -format F) Start logging"</code>	Tuesday, February 05, 2008 9:15:13 PM
<code>"\$(Get-Date -format o) Start logging"</code>	2008-02-05T21:15:13.0368750-05:00

There are many other formats for the `Get-Date` command, but these three options would generally suite most applications for timestamp purposes.

10. Stop and smell the roses

Within PowerShell, some commands have results that scroll through the screen very quickly. If you are not exporting the results to a file, it may be impossible to view the onscreen interaction. Let's again use the `Get-ChildItem` command from previous example. This command can return many results depending on your path contents. We'll create a function called `EasyView` to make it easy to view the results onscreen by displaying one line every half-second. The `EasyView` function would be created as follows:

```
function EasyView { process { $_; Start-Sleep -seconds .5}}
```

To make a PowerShell command use the `EasyView` function, call it with a pipe at the end of the command and then the function name as shown below:

```
Get-ChildItem N:\Data | EasyView
```

The `EasyView` function is configured to display lines at a half-second interval. You can also use milliseconds for the value.

Determine if your hardware can support Windows XP Mode in Windows 7

One of the most innovative features in Windows 7, from an IT perspective that is, is the new Windows XP Mode. This feature makes it easy to install and run Windows XP applications right from within Windows 7 by way of hardware-based virtualization technology and the new Windows Virtual PC utility — the successor to Microsoft Virtual PC.

I've wanted to experiment with Windows XP Mode in Windows 7 for quite some time, but only recently got around to accomplishing that objective. I now have Windows XP Mode up and running on my Windows 7 system and am truly amazed by the way that it works. It really is a very nice piece of work that should help alleviate the types of application compatibility issues that prevented Vista from gaining a strong foothold in the enterprise. Now, companies that rely on applications designed to run in Windows XP, will be able to move up to Windows 7 and still use their old applications.

As I began my initial exploration into what I would need to be able to run Windows XP Mode in Windows 7, I found myself stymied by apprehension, confusion, and misinformation related to the hardware requirements, as well as the lack of readily available instructions on how to get started. As you may know, there are a plethora of articles on the Internet decrying Intel's lack of support for virtualization technology in many of its CPU chips. This fact alone paints a daunting picture and thus fed my anxiety. Then there is the generalized notion that hardware-based virtualization is something that needs to be enabled in the BIOS before you can take advantage of it.

I'll help you cut through all the hype surrounding the hardware requirements for running Windows XP Mode in Windows 7 by describing the procedures that I went through to determine if my hardware would support Windows XP Mode.

My hardware exploration

While I am successfully running Windows XP Mode on my Windows 7 system, the road was fraught with trials and tribulations. I began my hardware exploration on the Microsoft Windows Virtual PC site, which outlines the system requirements for Windows Virtual PC as follows:

- Hardware Virtualization Technology enabled
- 1GHz 32-bit / 64-bit processor required
- Memory (RAM) 1.25GB required, 2GB memory recommended
- Recommended 15GB hard disk space per virtual Windows environment

My test system for this project started out as a bare-bones MSI box with an AMD Athlon 64 X2 3800+ CPU and 1GB of RAM. So, the first thing that I did was install another GB of

RAM. My processor was definitely fast enough and I had plenty of disk space. As such the next question was whether my AMD processor included AMD-V, which is the name of AMD's hardware virtualization technology.

When I originally built this system in December of 2006, hardware virtualization technology wasn't much of a blip on the radar, so I wasn't sure.

According to AMD:

“All CPUs AMD is currently shipping, except Sempron, include AMD-V and therefore support XP mode.”

In relation to its older CPUs, AMD stated:

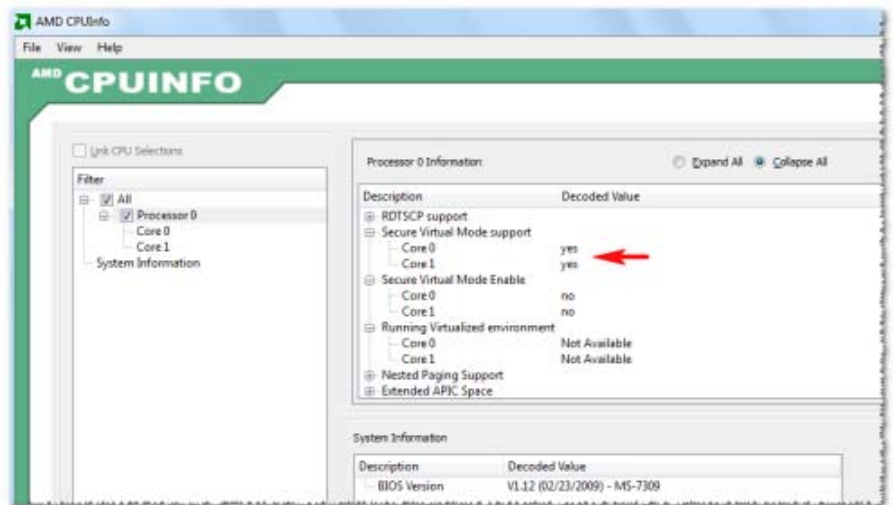
“With the exceptions of Sempron-branded processors and Turion K8 Rev E processors, all notebook processors shipped by AMD include AMD-V and therefore support Windows 7 XP mode. With the exceptions of Sempron-branded processors and pre-Rev F Athlon branded processors, all of the desktop processors shipped by AMD include AMD-V and therefore support Windows 7 in XP mode.”

While it would have been better if AMD had provided a list of all the processors that provided AMD-V support, based on this information, I was pretty confident that my AMD Athlon 64 X2 3800+ CPU supported AMD-V.

Assuming that I would have to enable it, I went into the system setup to look for a setting in the BIOS related to AMD-V but didn't find anything. However, I did notice that the BIOS was dated 10/19/2006. So, thinking that access to an AMD-V setting would be provided in a newer BIOS version, I took a trip to the MSI site and used the Live Update Online service to download a BIOS update. After flashing my BIOS, the new version date moved up to 2/23/2009, but there was still no setting related to AMD-V.

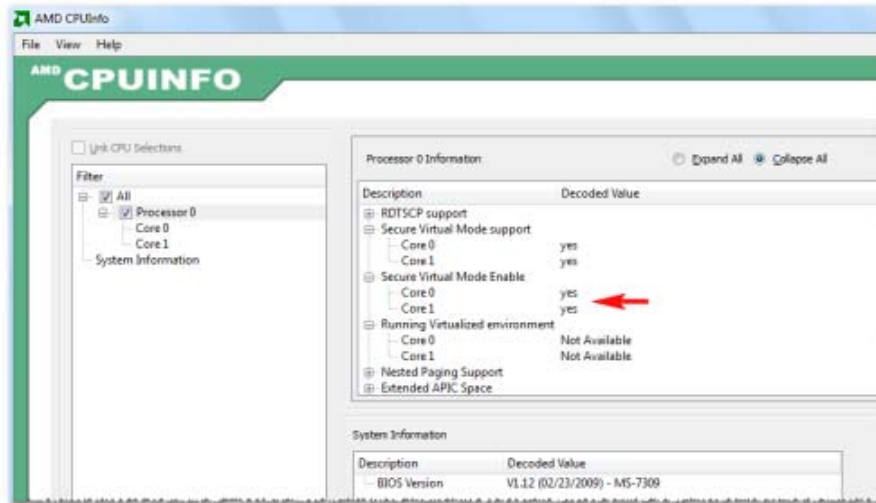
After a bit of searching on the AMD site, I found, downloaded, and installed the AMD CPUInfo utility. While it sort of provided me with the confirmation that my AMD Athlon 64 X2 3800+ CPU supported AMD-V, the information was conflicting. To begin with, instead of identifying the hardware virtualization technology as AMD-V, the CPUInfo utility identified it as Secure Virtual Mode, as shown in **Figure A**. Furthermore, while it verified that the CPU supported Secure Virtual Mode, it also indicated that Secure Virtual Mode was not enabled and as such, not available — more confusion.

Figure A



The AMD CPUInfo utility provided conflicting information.

Figure B



When Windows Virtual PC is running, the Secure Virtual Mode is listed as enabled.

However, Googling Secure Virtual Mode turned up a link to a January 2008 blog entry by John Howard who is a Senior Program Manager on the Hyper-V team, Windows Core Operating System Division. In this blog, John states:

“...On Intel platforms, you can normally turn VT on and off in the BIOS, whereas on AMD platforms, AMD-V is on all the time (at least I have not come across a BIOS which contains an option to enable/disable AMD-V).”

With this bit of wisdom in mind, I decided to just go ahead and install the beta version of Windows Virtual PC utility. It installed without a hitch. So I proceeded to install the beta version of Windows XP Mode and it too installed without a hitch.

Curious, I then ran the AMD CPUInfo utility again and discovered that while Windows Virtual PC is running, the Secure Virtual Mode is listed as enabled, as shown in **Figure B**. However, as soon as you close Windows Virtual PC, the Secure Virtual Mode is listed as not enabled. As such, it seems that AMD-V is designed to recognize when it is needed and will automatically enable itself.

Intel CPUsWhen it comes to using Windows XP Mode in Windows 7 on a system powered by an Intel CPU, I can't offer any firsthand experience with Intel VT at this time. I don't currently have an Intel system and haven't used one in quite some time. I'll see what I can come up with though.

However, according to articles, such as Ed Bott's recent Microsoft Report blog *How many Intel CPUs will fail the XP Mode test in Windows 7?*, over on ZDNet, there are a lot of Intel CPUs that do not provide hardware virtualization technology support. You can find out if your Intel CPU supports Intel VT by using the online Processor Spec Finder or by reviewing the lists on Ed's blog.

10 reasons why Windows 7's XP Mode is a big deal

One of the most exciting Windows 7 features is Windows XP Mode. It uses a brand new version of Virtual PC to provide seamless access to Windows XP applications, either through a virtual Windows XP desktop or directly through the Windows 7 desktop. Here's a look at some of the benefits XP Mode offers.

1. It solves compatibility problems

The biggest beef that most IT folks seem to have with Windows Vista is its notorious hardware and software compatibility problems. Windows 7's Windows XP mode allows you to run Windows XP applications without worrying about application compatibility.

2. It provides a much needed upgrade to Virtual PC

Virtual PC has been around for a long time, and although it has improved from one version to the next, it still leaves a lot to be desired. Among the improvements in the new version is the ability to access the computer's physical hard drives (including the host operating system's volumes) through a virtual machine.

3. It offers USB Support

Another much needed improvement to Virtual PC (which Windows XP Mode depends on) is that it now offers USB support. It has previously been impossible to access USB devices from within a virtual machine.

4. It's a way to modernize Windows XP

I know that there are those who would disagree with me, but Windows XP hasn't aged well. First introduced in 2001, Windows XP is quickly becoming outdated. Windows XP Mode provides enables you to run Windows XP inside a modern operating system, which helps it take advantage of some of the improvements that have been made to things like hardware support and security. Windows XP itself hasn't changed, but because Windows XP Mode is dependent on the host operating system, it can reap some of these benefits.

5. It ensures long-term technical support

Microsoft's continued support for Windows XP has been questionable for quite some time now. Every time Microsoft gets ready to pull the plug on main stream technical support, they give in to pressure from customers and extend the support period. It's great that Microsoft has been so accommodating, but nobody knows how long that will last. Having Windows XP Mode built into Windows 7 helps ensure that Windows XP support will be available for many years to come.

6. Microsoft has made a commitment to XP

For the last several years, Microsoft has urged customers to adopt Windows Vista, but most of Microsoft's corporate customers have chosen to continue using Windows XP. By including Windows XP mode in Windows 7, Microsoft has finally acknowledged the importance of Windows XP to its customers and given diehard XP fans a real solution that will allow them to move forward without giving up the OS they've depended on for almost a decade.

7. It offers seamless integration

One of my favorite things about Windows XP Mode is that it's completely seamless. Sure, you can work within a full-blown Windows XP virtual machine, but you don't have to. In fact, if you close the Windows XP virtual machine, you can access your Windows XP applications directly through the Windows 7 start menu and run those applications seamlessly alongside applications that are installed directly on Windows 7.

8. It's a first

This is the first time Microsoft has ever given us this type of support for an older product. Exchange 2000 included a copy of Exchange 5.5, but that was only included as part of the migration path for Exchange 5.0 users. Microsoft wasn't expecting customers to actually use both products. Making Windows XP part of the Windows 7 operating system is unprecedented.

9. It opens the door to lightweight operating systems

Windows has always had a bad reputation for being excessively bloated. One of the reasons for the bloat is that most versions of Windows have included a significant amount of code to provide backward compatibility with the previous version. By relying on virtualization to provide this compatibility, Microsoft may be able to greatly reduce the size of the core operating system in Windows 8.

10. Future plug-ins are possible

The way Microsoft has connected Windows XP to Windows 7 through virtualization opens the door to future operating system plug-ins. Don't be surprised if Windows 8 gives you the ability to pick and choose the legacy operating systems you want to support. Microsoft could end up offering virtualization plug-ins that will allow it to support Windows XP, Vista, and Windows 7. Using this method would allow customers to pick the type of backward compatibility they need without having to install any unnecessary legacy code.

Configure a USB flash drive to be a Windows 7 installation platform

In a recent 10 Things blog, “10 Cool Things You Can Do with a USB Flash Drive,” I presented 10 tasks that you can perform with a flash drive, including booting an operating system. However, I have since discovered a technique on the Microsoft TechNet Magazine site that describes how you can use a USB flash drive to install Windows 7.

This technique is very cool for two reasons: First, a flash drive is much more responsive than an optical drive, since it doesn't rely on physically moving components, so the installation procedure will be faster. Second, it provides an easy way to install Microsoft Windows 7 on a system that doesn't have a DVD drive, such as a netbook.

Since this technique would be a valuable addition to the 10 Things list as well as to users planning on installing Windows 7 on a netbook or who want to be able to quickly and easily install Windows 7, such as the Family Pack, on multiple systems without having to mess around with a DVD, I thought that I would examine the technique.

Getting started

Of course, in order to employ this technique you'll need a USB flash drive that is big enough to hold the contents of the Windows 7 DVD. The Windows 7 RC DVD is about 2.5GB, and you can expect that the RTM DVD will be at least that if not more. For my test system I am using a 4GB USB flash drive.

Keep in mind that the procedure we will use will completely reformat the USB flash drive. So you want to make sure that you back up any data that you have on the drive before you begin.

The systems on which you want to install Windows 7 via the USB flash drive have to be able to be configured to boot from a USB drive. Most new systems have the capability to boot from a USB flash drive, and the operation can usually be configured in the BIOS or by simply pressing a certain key during bootup. You'll need to check your specific hardware in order to be sure.

Using the DiskPart utility As you may know, the DiskPart utility is a command-line version of the Disk Management snap-in and is designed to allow you to manage disks, partitions, or volumes from within scripts or directly from a command prompt. We can use the DiskPart command in Windows Vista or in Windows 7 to configure a USB flash drive to be a bootable device. (Keep in mind that Windows XP's DiskPart command is unable to create a bootable USB flash drive.)

To begin, connect your USB flash drive to a computer on which you will be preparing the drive. For my example, I'll be using a Windows Vista system.

Now, locate the command Prompt shortcut on the Start menu, right-click on it, and select the Run as Administrator command. Then, respond appropriately to the UAC. You can now launch the DiskPart utility by typing DiskPart on the command line. You'll then see the DISKPART prompt.

Now that you're in the DiskPart environment, you'll need to locate the USB flash drive using the List Disk command. As you can see in **Figure A**, the List Disk command has identified my USB flash drive as Disk 5. I can verify that my USB flash drive is indeed Disk 5 by checking the Size column, which lists the size as 3906MB, which is roughly 4GB.

(If you have difficulty identifying your USB flash drive using the List Disk command, you can try the List Volume command, which will provide the drive letter as well as the label, both of which can help you to identify the drive.)

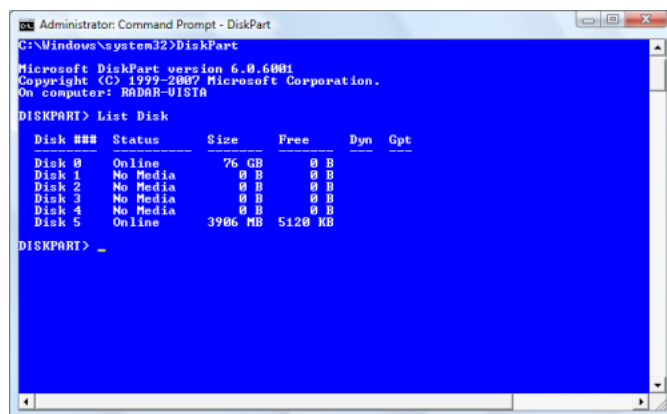
Once you identify the drive number of your USB flash drive, you will need to set the focus of the DiskPart environment on that disk. (This is an extremely important step—Make sure that you select the correct drive or you could accidentally destroy valuable data!). On my example, the USB flash drive is Disk 5, so I will use the command Select Disk 5, as shown in **Figure B**.

Now that your USB flash drive has the focus, you need to remove all the partition or volume formatting information from the disk. To do that, you'll use the Clean command. The Clean operation should occur rather quickly. When it is done you will see a success message like the one shown in **Figure C**.

You'll now use the Create Partition Primary command to create a primary partition on the disk. After you create the partition, you will see a success message and the focus will automatically shift to the new partition. You'll use the Active command to mark the partition as active. Marking the partition as active will essentially allow the BIOS to recognize that the partition is a valid bootable system partition.

With the partition created and active, you're now ready to quickly format the drive and set up the FAT32 file system using the Format fs=FAT32 quick command. (While you could format the drive as NTFS, the typical way to format a USB flash drive is to use FAT32.) Once the drive is formatted, you'll use the Assign command to allow the drive to be assigned a drive letter.

Figure A



```
Administrator: Command Prompt - DiskPart
C:\Windows\system32>DiskPart
Microsoft DiskPart version 6.0.6001
Copyright (C) 1999-2007 Microsoft Corporation.
On computer: RADAR-U15TA

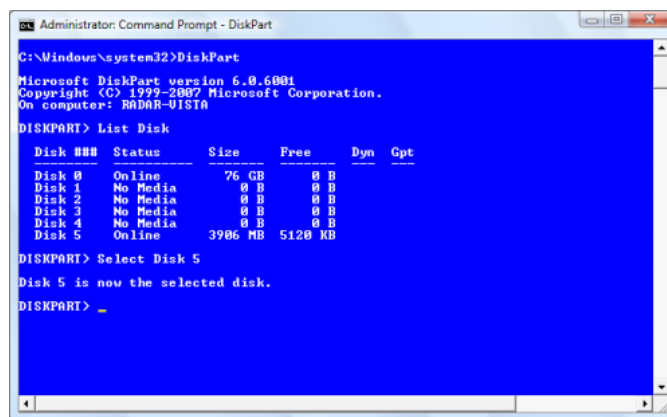
DISKPART> List Disk

Disk ### Status      Size      Free      Dyn  Gpt
-----
Disk 0  Online      76 GB     0 B       0    0
Disk 1  No Media    0 B       0 B       0    0
Disk 2  No Media    0 B       0 B       0    0
Disk 3  No Media    0 B       0 B       0    0
Disk 4  No Media    0 B       0 B       0    0
Disk 5  Online      3906 MB   5120 KB   0    0

DISKPART> -
```

Using the List Disk command displays all the disks in the system.

Figure B



```
Administrator: Command Prompt - DiskPart
C:\Windows\system32>DiskPart
Microsoft DiskPart version 6.0.6001
Copyright (C) 1999-2007 Microsoft Corporation.
On computer: RADAR-U15TA

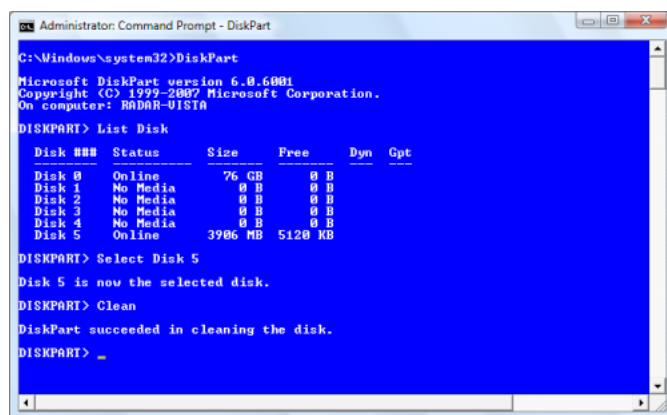
DISKPART> List Disk

Disk ### Status      Size      Free      Dyn  Gpt
-----
Disk 0  Online      76 GB     0 B       0    0
Disk 1  No Media    0 B       0 B       0    0
Disk 2  No Media    0 B       0 B       0    0
Disk 3  No Media    0 B       0 B       0    0
Disk 4  No Media    0 B       0 B       0    0
Disk 5  Online      3906 MB   5120 KB   0    0

DISKPART> Select Disk 5
Disk 5 is now the selected disk.
DISKPART> -
```

To shift the focus over to the USB flash drive you'll use the Select Disk command.

Figure D



```
Administrator: Command Prompt - DiskPart
C:\Windows\system32>DiskPart
Microsoft DiskPart version 6.0.6001
Copyright (C) 1999-2007 Microsoft Corporation.
On computer: RADAR-U15TA

DISKPART> List Disk

Disk ### Status      Size      Free      Dyn  Gpt
-----
Disk 0  Online      76 GB     0 B       0    0
Disk 1  No Media    0 B       0 B       0    0
Disk 2  No Media    0 B       0 B       0    0
Disk 3  No Media    0 B       0 B       0    0
Disk 4  No Media    0 B       0 B       0    0
Disk 5  Online      3906 MB   5120 KB   0    0

DISKPART> Select Disk 5
Disk 5 is now the selected disk.
DISKPART> Clean
DiskPart succeeded in cleaning the disk.
DISKPART> -
```

In order to start with a clean slate you'll use the Clean command to remove all partition and volume information from the USB flash drive.

The USB flash drive is now ready. At this point, you can use the Exit command to exit the DiskPart environment and then close the Command Prompt window.

Copying the Windows 7 files Copying the Windows 7 files is easy. Just open Windows Explorer, access the Windows 7 DVD, select all the files and folders, and then drag and drop them on the USB flash drive icon. Keep in mind that the copy operation will take a little while to complete.

Installing Windows 7 from the USB flash drive With the contents of the Windows 7 DVD on a bootable USB flash drive, installing the operating system is a snap. Just boot the system from the USB drive and the installation procedure will begin as it normally would. However, the installation procedure will actually run quicker off of a USB flash drive since it doesn't contain any physically moving components.

Once the system boots from the USB flash drive, the Windows 7 installation will begin as normal, but it will actually run faster.

Windows 7 now allows you to boot a native OS from a VHD file

Earlier this year, I configured my Dell Latitude E6500 laptop to dual boot from Windows X7 installed locally to Windows Server 2008 on an externally connected eSATA hard disk. This process works fine and is the traditional method of booting to a native environment from a PC.

New to Windows 7 is the ability to leverage an existing VHD file and boot to the environment contained in it natively.

In my case, I have several VHD files that work in Hyper-V, which I use as part of my Server 2008 installation. These are on my external eSATA drive, which is configured to display as drive V: to my Windows 7 native install (**Figure A**).

I have a virtual machine that uses Windows Vista from the VHD file displayed in **Figure B**.

OK, now let's see what I have as bootable options on my laptop. Before Vista, you needed to look at the boot.ini file. Windows Vista and Windows 7 use the bcdedit.exe utility. You will need to run the command prompt with elevated privileges to see it (Run As Administrator).

As you can see in **Figure C**, I have my Windows 7 environment and my Windows Server 2008 environment to choose from. I'll need to add an entry for the Windows Vista option to boot from VHD.

To do this, I'll copy my current environment entry and then edit the copied entry with the appropriate parameters. To do this, type:

```
bcdedit /copy {current} /d "Windows Vista"
```

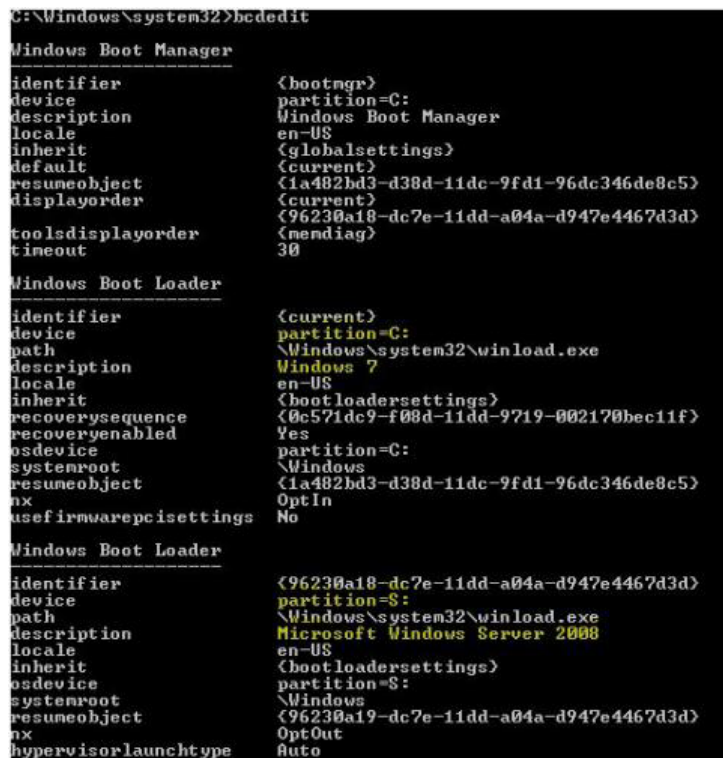
Figure A



Figure B



Figure C



My output is shown in **Figure D**:

The new entry is shown in **Figure E**:

Notice how right now, this is reflected as coming from drive C:, which will need to be changed. I now have a unique CLSID that can be associated as a new boot device, and my description reads correctly as Windows Vista.

Now, let's edit the newly added boot device with the correct values. In my case the correct path to the VHD file must be supplied and associated to the correct CLSID. To do this, type:

```
bcdedit /set {CLSID_number} osdevice  
vhd=[v:]"Hyper-V Virtual Hard Disks\  
WS03 VMAdd.vhd"
```

Note: Replace with your drive, path, and filename.

You are done!

To get a sample of what the boot menu will look like before you reboot, from the Run menu, launch `msconfig`. (**Figure F**)

Now, you can boot Windows Vista from a VHD file.

Figure D

```
C:\Windows\system32>bcdedit /copy {current} /d "Windows Vista"  
The entry was successfully copied to {0c571dcb-f00d-11dd-9719-002170bec11f}.  
C:\Windows\system32>
```

Figure E

```
Windows Boot Loader  
-----  
identifier          {96230a18-dc7e-11dd-a04a-d947e4467d3d}  
device              partition=S:  
path                \Windows\system32\winload.exe  
description         Microsoft Windows Server 2008  
locale              en-US  
inherit             <bootloadersettings>  
osdevice            partition=S:  
systemroot          \Windows  
resumeobject        {96230a19-dc7e-11dd-a04a-d947e4467d3d}  
nx                  OptOut  
hypervisorlaunchtype Auto  
-----  
Windows Boot Loader  
-----  
identifier          {0c571dcb-f00d-11dd-9719-002170bec11f}  
device              partition=C:  
path                \Windows\system32\winload.exe  
description         Windows Vista  
locale              en-US  
inherit             <bootloadersettings>  
recoverysequence    {0c571dc9-f00d-11dd-9719-002170bec11f}  
recoveryenabled      Yes  
osdevice            partition=C:  
systemroot          \Windows  
resumeobject        {1a482bd3-d38d-11dc-9fd1-96dc346de8c5}  
nx                  OptIn  
usefirmwarepcisettings No  
C:\Windows\system32>
```

Figure F

