

# Deploying System Images

This chapter discusses the management, manipulation, and deployment of system images. You *deploy* an image when you install it on one or more target computers. It looks at how you mount a system image so it can be updated and altered, how you then commit these changes to the original image, and how you distribute that image to a number—often a large number—of client computers. In the context of system images, you *mount* an image by expanding it into a folder so you can obtain information about it and add or remove features such as drivers, updates, and language packs.

Microsoft provides a range of tools for image manipulation and deployment, some of which are specific to Windows 7 images, whereas others are more general. Some tools manipulate images, others deploy them, and some tools do both. It is your job as a network administrator to choose the best tools for your current and future needs and configure them so that they work efficiently and go on working efficiently.

If your users are sitting at computers with operating systems, drivers, and applications that are up to date and as invulnerable as you can make them to Internet (and intranet) attacks they will be less unhappy. If you can bring one computer or 100 computers into full operation quickly, efficiently, and without error, then your boss will be less unhappy. (We all know, of course, that neither users nor bosses are ever happy.)

## Exam objectives in this chapter:

- Prepare a system image for deployment.
- Deploy a system image.

## Lessons in this chapter:

- Lesson 1: Managing a System Image Before Deployment **116**
- Lesson 2: Deploying System Images **146**

## Before You Begin

To complete the exercises in the practices in this chapter you need to have done the following:

- Installed the Windows 7 operating system on a stand-alone client PC, as described in Chapter 1, “Install, Migrate, or Upgrade to Windows 7.” You need Internet access to complete the exercises.
- Completed all the practice exercises in Chapter 2, “Configuring System Images.” In particular, you need to have installed the Windows Automated Installation Kit (Windows AIK) and deployed an offline image of the Canberra computer on a bootable virtual hard disk (VHD).



### REAL WORLD

Ian McLean

Each of the tools you use for network and system administration has its own set of features and enables you to perform specific tasks. Where people sometimes get upset and confused is if there is overlap. For example, you can use Windows Deployment Services (WDS) or the Microsoft Deployment Toolkit (MDT) 2010 to deploy Windows 7 images to client computers. However, MDT 2010 allows you to specify a set of configuration tasks that should be run on a computer after an image has been deployed to it through WDS, whereas WDS cannot run configuration tasks on a client after the image has been deployed.

You use ImageX to create an image of a computer running Windows 7 while it is booted to Windows Preinstallation Environment (Windows PE) and you use the Deployment Image Servicing and Management (DISM) tool to manipulate that image after it has been created. However, you can use ImageX or DISM to mount an image so you can work with it offline.

So, you are entitled to ask, which tool do I use and, more to the point in this book, what tool will the examination ask about? The simple answer is to use the most recently introduced tool when there is a choice. For example, ImageX has been around for some time, whereas DISM was introduced fairly recently; however, ImageX has new features in the latest edition of the Windows AIK.

Traditionally, examinations ask more questions about new features of an operating system and new tools that are introduced to carry out tasks on the new operating system than they do about features that are unchanged from the previous operating system and tools that, however worthy, have been around for a while. This is simply a statement of fact and any conclusions you draw from it are your own.

However, as a professional administrator rather than merely an examination candidate, what tool do you choose to do your job? That's up to you. My advice is if you've been using DISM frequently in the recent past and have got into the rhythm of typing in DISM commands, use that tool if you are given the choice. Otherwise, my preferred tool when I'm mounting an image is ImageX. The commands are shorter (important when you're as slow a typist as I am), I find the command completes sooner, and if something does go wrong, ImageX usually gives a more detailed explanation. That, however, is only my opinion, with which you are absolutely entitled to disagree.

As a professional, use your best judgment. As an examination candidate, know what all the tools do, what a tool does that no other tool can do, what a tool does that is unique to the new operating system or that you haven't been able to do before, and what jobs can be done equally well by two or more tools.

Good luck.

## Lesson 1: Managing a System Image Before Deployment

Sometimes when you have created a master reference image for deploying to other computers, you might find that you need to amend it. You might need to add a new driver, change settings, and support multiple languages. Even a fairly minor change, such as enabling a feature currently disabled in the image, can generate a considerable workload if it needs to be done after the image has been distributed to several hundred computers.

Typically, it involves less administrative effort if you make these changes without deploying the image and recapturing it. If your requirement is to add security updates, then it is certainly preferable to apply the security patches offline—otherwise, you are deploying an insecure image. If you service the image offline, you do not need to run the Sysprep tool and therefore do not need to use a rearm parameter. Finally, you might want to apply an amended Autounattend.xml file for unattended install or an additional Unattend.xml file that automates post-installation tasks such as installing mission-critical applications. Unattend.xml files are discussed in the section entitled “Unattended Servicing Command-Line Options,” later in this chapter.

This lesson discusses how you use Windows AIK tools such as ImageX and DISM to mount a system image and how you use DISM to manage the image, insert packages, insert updates, enable and disable features, manage international settings, manage language packs, and associate unattend answer files.

### After this lesson, you will be able to:

- Mount an offline image for servicing.
- Use DISM to manage and manipulate the image.
- Associate one or more answer files with the image.
- Commit and unmount the image.

**Estimated lesson time: 50 minutes**

## Using DISM WIM Commands and Mounting an Image

Chapter 2 introduced the Windows 7 DISM command-line tool. DISM enables you to service offline images, mount and dismount Windows Imaging format (WIM) files, and customize Windows PE boot images. The DISM tool replaces many of the tools in previous versions of the Windows AIK, including Package Manager (Pkgmgr.exe), the International Settings Configuration Tool (Intlcfg.exe), and the Windows PE command-line tool (PEimg.exe).

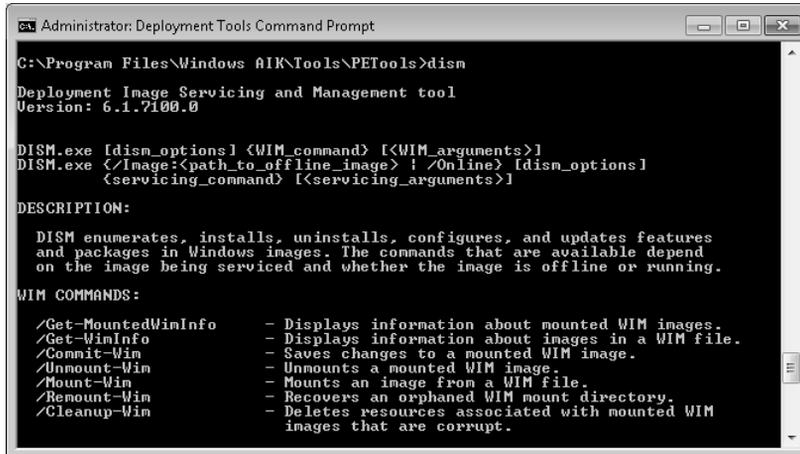
Microsoft has designed the DISM tool to manage WIM images. Also, DISM is backward-compatible with Vista tools, such as Pkgmgr.exe, Intlcfg.exe, and PEimg.exe, so scripts that you developed and tested for Vista should work unamended in Windows 7.

#### NOTE BACKWARD COMPATIBILITY

You can install the Windows AIK in Windows 7, including DISM, on Windows Vista SP1. If you manage Windows 7 and Windows Server 2008 R2 images under Vista, this backward-compatibility provides full functionality. DISM also works for Windows Vista SP1 and Windows Server 2008 images, but only with limited functionality. You can use all of DISM's features for Windows Vista SP1 images that would have worked with Pkgmgr.exe, Intlcfg.exe, or PEimg.exe. You will get an error message if you use a command-line option that is not supported for a Vista image.

Chapter 2 also introduced the ImageX tool, which you used to create a WIM image file. Typically you can use either DISM or Image X to mount WIM files. DISM is the recommended Windows AIK tool for managing and manipulating offline images. It allows you to install and configure operating system updates, packages, and drivers on an offline system image. You can use DISM to modify Windows PE images offline and to change the language, locale, fonts, and input settings on a Windows image.

The commands that DISM offers for image management depend upon the type of image you want to manage. You access DISM by clicking All Programs on the Start menu, clicking Microsoft Windows AIK, right-clicking Deployment Tools Command Prompt, and choosing Run As Administrator. You might need to click Yes to allow the program to run. Entering **dism** in the elevated Deployment Tools Command Prompt window generates a list of DISM commands. Figure 3-1 shows commands specific to WIM images.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism
Deployment Image Servicing and Management tool
Version: 6.1.7100.0

DISM.exe [dism_options] <WIM_command> [<WIM_arguments>]
DISM.exe </Image:<path_to_offline_image> | /Online> [dism_options]
        <servicing_command> [<servicing_arguments>]

DESCRIPTION:
    DISM enumerates, installs, uninstalls, configures, and updates features
    and packages in Windows images. The commands that are available depend
    on the image being serviced and whether the image is offline or running.

WIM COMMANDS:
    /Get-MountedWimInfo - Displays information about mounted WIM images.
    /Get-WimInfo - Displays information about images in a WIM file.
    /Commit-Wim - Saves changes to a mounted WIM image.
    /Unmount-Wim - Unmounts a mounted WIM image.
    /Mount-Wim - Mounts an image from a WIM file.
    /Remount-Wim - Recovers an orphaned WIM mount directory.
    /Cleanup-Wim - Deletes resources associated with mounted WIM
    images that are corrupt.
```

FIGURE 3-1 DISM commands for managing WIM images

For example, to get information about Myimage.wim, the WIM file that you created in Chapter 2 and installed on a bootable VHD with the drive letter W:, you enter the following command:

```
dism /get-wiminfo /wimfile:w:\myimage.wim
```

The output from this command is shown in Figure 3-2. If this command cannot find the image, open Computer Management and attach the VHD. It is the file Myvhd.vhd in the VHDs folder. If you did not create a VHD, you should find the same file in the D:\Images folder (or whatever destination you specified in the practice session in Chapter 2). If the file Myimages.wim does not exist at all, copy the file Install.wim from your installation DVD-ROM to D:\Images and use it instead.

```

Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /get-wiminfo /wimfile:w:\myimage.wim
Deployment Image Servicing and Management tool
Version: 6.1.7100.0

Details for image : w:\myimage.wim

Index : 1
Name : Brisbane Win7 Install
Description : <undefined>
Size : 30,125,462,460 bytes

The operation completed successfully.

C:\Program Files\Windows AIK\Tools\PETools>_
  
```

**FIGURE 3-2** Using the /Get-WimInfo and /Wimfile switches

However, if you want more detailed information about the image, you can use the ImageX tool as follows:

```
imagex /info w:\myimage.wim
```

Part of the output from this command is shown in Figure 3-3. The file contains only one image with the index number 1. (If you are using the Install.wim image, you need to discover the index number for the Windows 7 Ultimate image—typically 5.)

```

Administrator: Deployment Tools Command Prompt
WIM Information:
Path:          w:\myimage.wim
GUID:         {a21b7fce-ead1-4547-adc2-aa5df606ab19}
Image Count:  1
Compression:  XPRESS
Pare Number:  1/1
Attributes:   0x8
              Relative path junction

Available Image Choices:
<WIM>
<TOTALBYTES>13746048760</TOTALBYTES>
<IMAGE INDEX>"1"
<DIRCOUNT>19829</DIRCOUNT>
<FILECOUNT>116353</FILECOUNT>
<TOTALBYTES>30125462460</TOTALBYTES>
<HARDLINKBYTES>3531588373</HARDLINKBYTES>
<CREATIONTIME>
  <HIGHPART>0x01C9B42F</HIGHPART>
  <LOWPART>0x2A6D51E6</LOWPART>
</CREATIONTIME>
<LASTMODIFICATIONTIME>
  <HIGHPART>0x01C9B42F</HIGHPART>
  <LOWPART>0x2DDE8F0C</LOWPART>
</LASTMODIFICATIONTIME>
<WINDOWS>
  
```

**FIGURE 3-3** Detailed information about a WIM image

## Mounting WIM Images

If you want to manage an image, you first need to mount it. To mount the `W:\Myimage.wim` image on (for example) the `C:\MountedImages` folder, you enter the following command in the elevated Deployment Tools command prompt:

```
dism /mount-wim /wimfile:w:\myimage.wim /index:1 /mountdir:c:\mymountedimages
```

The `/index` flag in this command indicates the image that you want to mount. You can have several images (for example, several editions of Windows 7) within a single WIM file.

You can add the `/readonly` flag to this command if you want the image to be read-only. Alternatively, you can use the ImageX tool from the same console by entering the following command:

```
imagex /mountrw w:\myimage.wim 1 c:\mountedimages
```

If you want to try out both commands, you need to create and use another destination folder for the second command (for example, `C:\Othermount`). The destination folder must initially contain no files. Alternatively, you can delete the mounted image in the `C:\MountedImages` folder and regenerate the mounted image. You cannot use Windows Explorer to delete a mounted image but must instead unmount it by using the following ImageX command:

```
imagex /unmount c:\mountedimages
```

You can also use DISM to unmount a mounted WIM file by entering a command similar to the following:

```
dism /unmount-wim c:\mountedimages
```

You use these commands in the practice session later in this lesson.

### **NOTE MOUNTING THE INSTALL.WIM FILE ON THE INSTALLATION DVD-ROM**

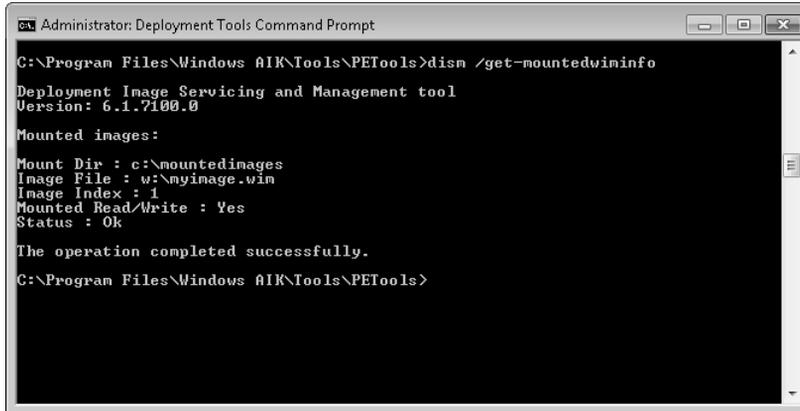
You can mount the `Install.wim` file on the installation DVD-ROM, but this enables you to create only a read-only image, and you need to specify the `/readonly` flag if you execute a DISM command. If you want to create an image that you can manipulate, you need first to copy `Install.wim` to rewritable media and ensure that the Read Only file attribute is not set.

## Getting Information About Mounted WIM Images

You can get information about all mounted WIM files on your computer by entering the following command:

```
dism /get-mountedwiminfo
```

The output from this command is shown in Figure 3-4.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /get-mountedwiminfo
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Mounted images:
Mount Dir : c:\mountedimages
Image File : w:\myimage.wim
Image Index : 1
Mounted Read/Write : Yes
Status : Ok
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>
```

FIGURE 3-4 Mounted WIM files on the computer

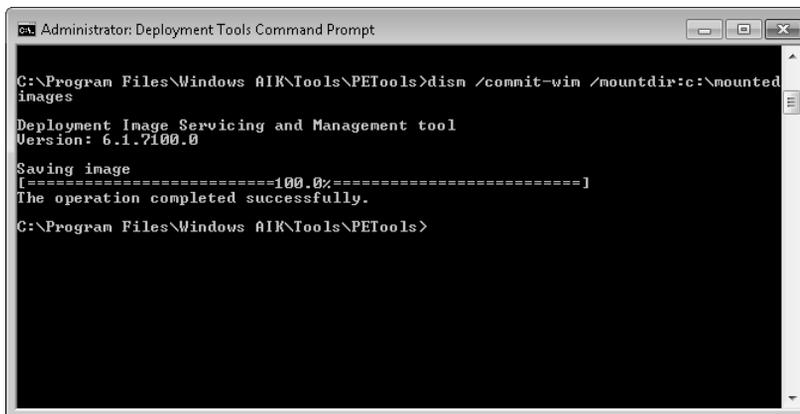
If you are working with files from a mounted image, it is good practice to create a temporary scratch directory in which you can place these files. You need first to create a folder to hold the files (for example, C:\Working Files). You then enter the following command:

```
dism /image:c:\mountedimages /scratchdir:c:\workingfiles
```

When you have created a mounted Read/Write image, you can amend and update existing files and add applications and drivers. You can specify edition and international settings and add language packs. These operations are described later in this lesson. You can save (or *commit*) any amendments you make to the offline image by using a command similar to the following:

```
dism /commit-wim /mountdir:c:\mountedimages.
```

This command can take a considerable time to complete. In this example, it saves any changes you have made to the mounted image in C:\MountedImages to the source image W:\Myimage.wim. The command does not unmount the image and you have the option of further amending and saving the image. Figure 3-5 shows the output from this command.



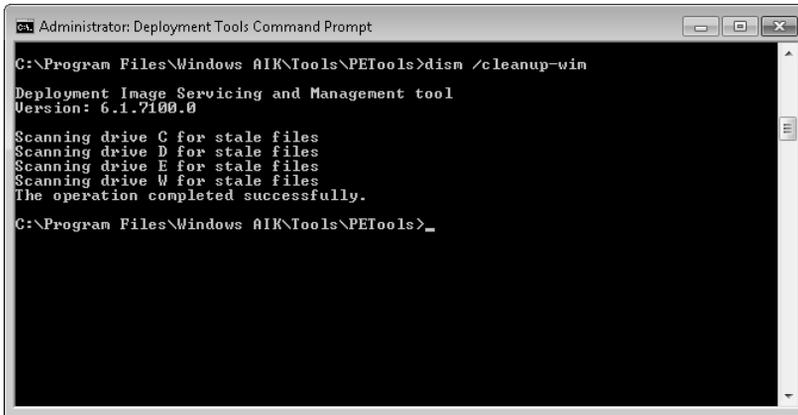
```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /commit-wim /mountdir:c:\mounted
images
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Saving image
[=====100.0%=====]
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>
```

FIGURE 3-5 Saving an amended mounted WIM image

Files in a mounted WIM image could become corrupt and you would not want to save such files back to the source image. It is also possible that a mounted image could become orphaned because of changes in directory structure. You can remove corrupt files from all mounted images on the computer with the following command:

```
dism /cleanup-wim
```

Figure 3-6 shows that all writable volumes are scanned for corrupt files.



**FIGURE 3-6** Scanning for corrupt files

You can retrieve and remount orphaned images by entering a command similar to the following:

```
dism /remount-wim /mountdir:c:\mountedimages
```



**EXAM TIP**

Distinguish between */cleanup-wim*, which removes corrupt files, */remount-wim*, which retrieves and remounts orphaned images, and */cleanup-image*. This last option is typically used with the */RevertPendingActions* parameter to attempt a system recovery if you experience a boot failure. This operation reverts all pending actions from the previous servicing operations because these actions might be the cause of the boot failure. Note that */RevertPendingActions* is not supported on a running operating system or a Windows PE or Windows Recovery Environment (Windows RE) image.

## Working with an Online Image

In addition to mounting and manipulating an offline image, you can work with the operating system image that is currently online. For example, the following command lists all the out-of-box drivers that are currently installed:

```
dism /online /get-drivers
```

Figure 3-7 shows some of the output from this command. If you want to list all drivers rather than all out-of-box drivers, append `/all` to the command.

```

Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /online /get-drivers
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Image Version: 6.1.7100.0
Obtaining list of 3rd party drivers from the driver store...
Driver packages listing:
Published Name : oem0.inf
Original File Name : prnms001.inf
Inbox : No
Class Name : Printer
Provider Name : Microsoft
Date : 6/21/2006
Version : 6.1.7100.0
Published Name : oem1.inf
Original File Name : hpqlb.inf
Inbox : No
Class Name : HIDClass
Provider Name : Hewlett-Packard
Date : 3/30/2007
Version : 4.20.5.0
  
```

**FIGURE 3-7** Listing drivers in the online system image

In general, you get information from an online image rather than amend or manipulate it. For example, a command such as `dism /online /get-currentedition`, which returns the edition of the operating system, could be used in a batch file where the action implemented depends upon the Windows 7 edition. Table 3-1 shows parameters that can be used with the `/Online` option.

**TABLE 3-1** Parameters That Can Be Used Online

PARAMETER	DESCRIPTION
<code>/Get-CurrentEdition</code>	Displays the edition of the online image
<code>/Get-StagedEditions</code>	Displays a list of Microsoft Windows editions that can be removed from an image
<code>/Get-TargetEditions</code>	Displays a list of Windows editions to which the online image could be upgraded
<code>/Get-DriverInfo</code>	Displays information about a specific driver
<code>/Get-Drivers</code>	Displays information about all out-of-box drivers
<code>/Get-Intl</code>	Displays information about the international settings and languages
<code>/Get-Packages</code>	Displays information about all packages in the online image
<code>/Get-PackageInfo</code>	Displays information about a specific package
<code>/Get-Features</code>	Displays information about all features in the online image
<code>/Get-FeatureInfo</code>	Displays information about a specific feature

Any parameter that can be used online can also be used offline by specifying a mounted WIM image with the */image* switch. For example, the following command lists all the drivers in the image mounted in the folder C:\MountedImages:

```
dism /image:c:\mountedimages /get-drivers /all
```

Table 3-2 lists the information retrieval parameters that you can use with an offline mounted image but not with an online image.

**TABLE 3-2** Parameters That Cannot Be Used with an Offline Image

PARAMETER	DESCRIPTION
/Get-AppPatchInfo	Displays information about installed Windows Installer patch files (MSP patches)
/Get-AppPatches	Displays information about all applied MSP patches for all installed applications
/Get-AppInfo	Displays information about a specific installed Windows Installer (MSI) application
/Get-Apps	Displays information about all installed MSI applications

## Servicing Drivers, Applications, Patches, Packages, and Features

You can use driver servicing commands on an offline mounted image to add and remove drivers based on the .inf file format. You can specify a directory where the driver .inf files are located, or you can point to a driver by specifying the name of the .inf file.

On an online running operating system, you can only enumerate drivers and obtain driver details. The commands and options to list drivers and obtain driver information were discussed in the previous section of this lesson. DISM can manage only .inf file drivers. Windows Installer (MSI) and other driver package types (such as .exe files) are not supported.

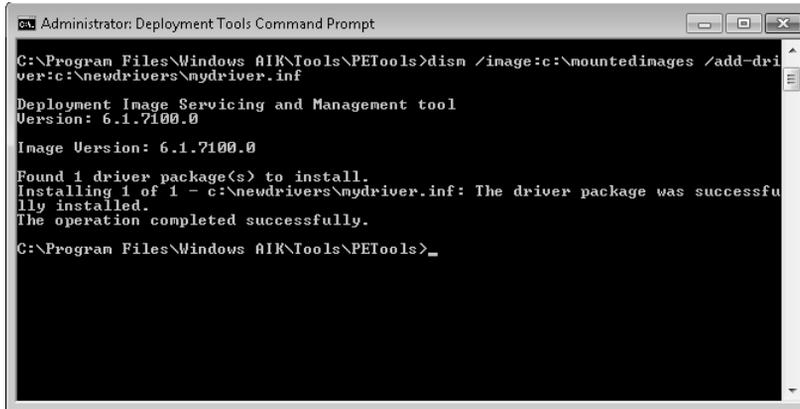
The following driver servicing options are available for an offline image:

```
dism /image: path_to_image_directory [/get-drivers | /get-driverinfo | /add-driver | /remove-driver]
```

For example, if you wanted to add the driver Mydriver.inf that you have downloaded and stored in the folder C:\Newdrivers, you would use a command similar to the following:

```
dism /image:c:\mountedimages /add-driver:c:\newdrivers\mydriver.inf
```

Figure 3-8 shows the output from this command.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /image:c:\mountedimages /add-driver:c:\newdrivers\mydriver.inf
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Image Version: 6.1.7100.0
Found 1 driver package(s) to install.
Installing 1 of 1 - c:\newdrivers\mydriver.inf: The driver package was successfully installed.
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>_
```

**FIGURE 3-8** Adding a driver to an image

When you are adding a new driver, you should store it in a convenient location. In this case, you can specify the file name and path directly after the Add-Drivers option and do not need the */driver* parameter. However, if you want to add several drivers that are in the same folder you would enter a command similar to the following:

```
dism /image:c:\mountedimages /add-driver /driver:c:\newdrivers
```

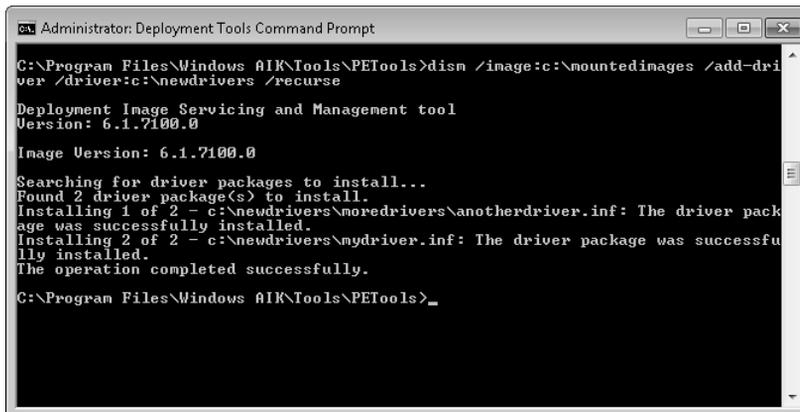
If you want to add all the drivers in two or more folders, you can use the */driver* parameter as many times as you require; for example:

```
dism /image:c:\mountedimages /add-driver /driver:c:\printdrivers /driver:d\nicdrivers.
```

If you want to add all the drivers in a folder and its subfolders you can use the */recurse* option. For example:

```
dism /image:c:\mountedimages /add-driver /driver:c:\newdrivers /recurse
```

Figure 3-9 shows the output from this command.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /image:c:\mountedimages /add-driver /driver:c:\newdrivers /recurse
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Image Version: 6.1.7100.0
Searching for driver packages to install...
Found 2 driver package(s) to install.
Installing 1 of 2 - c:\newdrivers\moredrivers\anotherdriver.inf: The driver package was successfully installed.
Installing 2 of 2 - c:\newdrivers\mydriver.inf: The driver package was successfully installed.
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>_
```

**FIGURE 3-9** Using the */recurse* option

On x64-based computers running Windows 7, drivers must have a digital signature. However, you might want to install an unsigned driver for test purposes. In this case you can use the */forceunsigned* parameter to override this requirement.

You can use the */remove-drivers* option to remove third-party drivers from an offline image. You cannot remove default drivers with the *dism* command. When you add third-party drivers, they are named *oem0.inf*, *oem1.inf*, and so on. You must specify the published name, but fortunately, the */get-drivers* parameter lists both the published name and the original name. If you have installed a lot of third-party drivers and are having difficulty finding the new name of the driver you want to remove, you can direct the output from a *dism* command that uses the */get-drivers* parameter to a text file and search this file for the original name. When you have identified the driver's published name, such as *oem10.inf*, you can then remove it using a command similar to the following:

```
dism /image:c:\mountedimages /remove-driver /driver:oem10.inf
```

## Servicing Applications and Application Patches

You can use application servicing command-line options applied to a offline image to check the applicability of Windows Installer application patches and to query the offline image for information about installed Windows Installer applications (.msi files) and application patches (.msp files).

None of the application servicing commands can be applied to online images, although if an image is online, it can receive updates from, for example, Windows Server Update Services (WSUS) or Microsoft Update. If you are administering an enterprise network, you should consider the Offline Virtual Machine Servicing Tool and System Center Virtual Machine Manager, which were discussed in Chapter 2.

DISM offers the following options available to list Windows Installer (.msi) applications and .msp application patches, and to check the applicability of an application patch on an offline system image:

```
dism /image:path_to_directory [/check-apppatch | /get-apppatchinfo: | /get-apppatches | /get-appinfo | /get-apps]
```

When managing applications and patches, your first step should be to discover what application patches and applications exist and are applicable to the image. For example, in an image mounted directly from an *Install.wim* file copied from the installation media, it is likely that no applicable patches or applications exist. To obtain information about application patches (MSI files) applicable to a mounted image, you would use a command similar to the following:

```
dism /image:c:\mountedimages /get-apppatches
```

If you know the product code globally unique identifier (GUID) of a Windows Installer application, you can use the */productcode* parameter to display all the application patches in the specified application. You would use a command similar to the following:

```
dism /image:c:\mountedimages /get-apppatches /productcode:{GUID}
```

If you want to display information about specific .msp patches applicable to the offline image, you can use the `/check-apppatch` parameter. You use `/patchlocation` to specify the path to the MSP patch file. You can specify multiple patch files by using `/patchlocation` more than once in the command. For example, to display information about two patch files, 30880d0.msp and 8c82a.msp (both in C:\Windows\Installer) in the mounted image, you would enter the following command:

```
dism /image:c:\mountedimages /check-apppatch
/patchlocation:c:\windows\installer\30880d0.msp
/patchlocation:c:\windows\installer\8c82a.msp
```

If you need detailed information about all installed MSP patches applicable to the offline image, you would enter a command similar to the following:

```
dism /image:c:\mountedimages /get-apppatchinfo
```

### ✓ Quick Check

- You want to add all the drivers in the folder C:\Orinsnewdrivers and its subfolders to the mounted offline image in D:\Orinsimage. What command would you use?

#### Quick Check Answer

- `dism /image:d:\orinsimage /add-driver /driver:d:\orinsimage /recurse`

You can use the `/get-apppatches` option described earlier in this section to find the patch code GUID and the product code GUID specific to a patch. You can also use the `/get-apps` option described here to list all product code GUIDs for an installed Windows Installer application. You can filter the information returned by the `/get-apppatchinfo` parameter either by the patch code GUID or the product code GUID, or by both, for example:

```
dism /image:c:\mountedimages /get-apppatchinfo /patchcode:{patch_code_GUID}
/productcode:{product_code_GUID}
```

In addition to obtaining information about applicable application patches, you typically need to obtain information about the MSI applications. The `/get-apps` parameter lists the MSI applications installed on the mounted image and you can use it to determine each application's GUID; for example:

```
dism /image:c:\mountedimages /get-apps
```

You can then obtain more detailed information about installed applications by using the `/get-appinfo` parameter. Optionally you can filter this information by specifying the product code GUID for the application in which you are interested; for example:

```
dism /image:c:\mountedimages /get-appinfo /ProductCode:{product_code_GUID}
```

If you do not specify a product code GUID, the `/get-appinfo` parameter returns detailed information about all installed MSI applications.



### EXAM TIP

Remember that `/get-apppatches` and `/get-apppatchInfo` apply only to installed patches (.msp files) and that `/get-apps` and `get-/appinfo` apply only to Windows Installer applications (.msi files). You cannot, for example, use DISM to obtain information about .exe or .dll files. Also, remember that when you check the applicability of an MSP patch, only the Windows Installer applications for which the patch is applicable will be displayed. One patch can be applied to many installed applications and many patches can be applied to one application.

## Adding Applications to an Image

The `/check-apppatch`, `/get-apppatchinfo`, `/get-apppatches`, `/get-appinfo`, and `/get-apps` DISM options obtain information about Windows Installer applications and installed patches on an offline mounted image. The next section describes how you add cabinet (.cab) or Windows Update Stand-alone Installer (.msu) files to an image and, in particular, install security patches to offline-mounted images. You can also enable and disable Windows features, but you cannot add features or any other type of executable files, such as .exe, .bat, .com, or .vbs files. The DISM command does not have an `/add-apps` option.

If you want to add a mission-critical application to the image for distribution, you can install that application on your reference computer before you image it. If, however, you want to add an application to an already existing offline image, the DISM tool does not do this. Instead, you should use the Add Application Wizard provided by MDT 2010, which is discussed in Lesson 2, “Deploying System Images.”

You can also use DISM to associate an image with an Unattend.xml answer file. Such a file automates installation of the image but also automates post-installation tasks, for example, connecting to a file server and installing applications or configuring settings. This approach, where applications and settings are applied after installation rather than included in the image, is known as “thin image” and is described in Lesson 2 of this chapter.

## Servicing Operating System Packages

One of the problems you have with system images either held for distribution to a number of computers or installed on a bootable VHD of a single client computer for failover purposes is that you need to keep the image up to date, particularly with security updates and fixes. Otherwise, if you boot with the new image, the computer is vulnerable to known security threats.

In the case of a single client computer where you have captured a system image as described in Chapter 2 and installed this to bootable VHD, you have several options. The most straightforward way, if possibly not the fastest, is to boot the computer from its VHD, go immediately to Microsoft Update, and install any critical or recommended packages. You can do this manually or, in an enterprise, by using the Offline Virtual Machine Servicing Tool on a server that has Microsoft System Center Virtual Machine Manager (SCVMM) installed. The disadvantage is that computers are brought online without the latest security updates and are vulnerable, if only for a short time.

You can also re-image your client computer on a regular basis and install the new image on to the VHD. This involves booting the computer to Windows PE and is a time-consuming process, especially when carried out on a regular basis. It has the advantage that the new image contains any software applications and user files that have been added to the computer since the last image capture and provides a form of backup.

If you have created an image for distribution to significant numbers of new computers, you can ensure your reference computer (as defined in Chapter 1) is kept fully up to date and create a fresh image whenever you want to configure a batch of client computers. You can also use MDT 2010 to manipulate images, as described in Lesson 2 of this chapter.

However, possibly the quickest and easiest way to manipulate images and install packages is to use DISM. DISM enables you to list Windows packages installed on a mounted offline image, get information about them, add and remove packages, and manipulate Windows features on a mounted offline image. You can also use DISM with an online operating system to perform the same operations, although if you want to ensure that an online image has all its critical and recommended updates installed, it is easier to use Microsoft Update or WSUS.

### ✓ Quick Check

- You want to display information about two patch files, `Mypatch.msp` and `Otherpatch.msp`, both in `C:\Windows\Patches`, in an image mounted in `D:\Myimages\Mountedimage1`. What command would you use?

### Quick Check Answer

- `dism /image:d:\myimages\mountedimage1 /check-apppatch /patchlocation:c:\windows\patches\mypatch.msp /patchlocation:c:\windows\patches\otherpatch.msp`

You can use DISM package-servicing commands with an offline-mounted image to install, remove, or update Windows packages provided as cabinet (.cab) or Windows Update Stand-alone Installer (.msu) files. Microsoft uses packages to distribute software updates, service packs, and language packs, and packages can also contain Windows features (optional features for the core operating system). You can use package-servicing commands to enable or disable Windows features both on an offline-mounted image and on a running Windows installation.

You can identify a package in your online image and install it on your mounted offline image. You can also disable and re-enable a feature.

For an offline image, you can use the following operating system package-servicing options:

```
dism /image:path_to_image_directory [/get-packages | /get-packageinfo | /add-package | /remove-package ] [/get-features | /get-featureinfo | /enable-feature | /disable-feature ]
```

For an online (running) operating system, you can use the following operating system package-servicing options:

```
dism /online [/Get-Packages | /Get-PackageInfo | /Add-Package | /Remove-Package] [/Get-Features | /Get-FeatureInfo | /Enable-Feature | /Disable-Feature]
```

If you want to compare an online operating system with an offline-mounted image, you need to first list the packages and features installed in both images. You would enter commands similar to the following:

```
dism /online /get-packages > c:\onlinepackages.txt
dism /image:c:\mountedimages /get-packages > c:\offlinepackages.txt
dism /online /get-features > c:\onlinefeatures.txt
dism /image:c:\mountedimages /get-features > c:\offlinefeatures.txt
```

It is a good idea to redirect the output of each of these commands to a text file. This enables you to compare lists easily. Also, the names of some packages can be long and complex, and it is useful to be able to copy them and paste them into the command line.

**NOTE FEATURE NAMES ARE CASE-SENSITIVE**

DISM commands are not case-sensitive. However, feature names are.

Suppose, for example, you wanted to find out more about the file *Package\_for\_KB970419~31bf3856ad364e35~x86~6.1.1.0*. You would enter a command similar to the following:

```
dism /online /get-packageinfo
/packageName:Package_for_KB970419~31bf3856ad364e35~x86~6.1.1.0
```

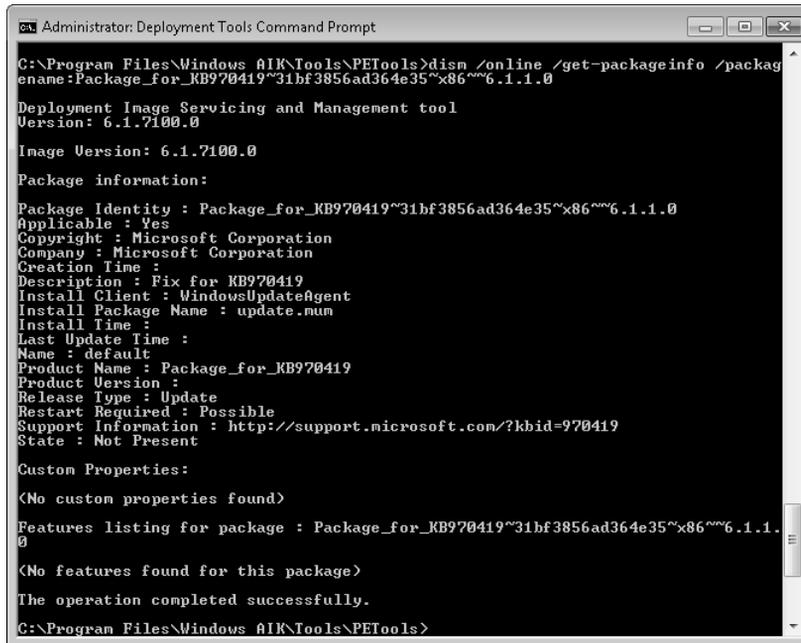
The output of this command is shown in Figure 3-10.

You cannot use the */get-packageinfo* option to get information about .msu files and you can specify only .cab files.

Suppose that you have downloaded or created a file called *Mypackage.cab* and placed it in a folder called *C:\Mypackages*. If you decide you need to insert this package into the offline mounted image, you would enter a command similar to the following:

```
dism /image:c:\mountedimages /add-package /packagepath:c:\mypackages\mypackage.cab
```

DISM checks the applicability of each package. If the package is not applicable to the specified image, DISM generates an error message. If you want the command to process without checking the applicability of each package, you can append the */ignorecheck* parameter.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /online /get-packageinfo /packag
ename:Package_for_KB970419~31bf3856ad364e35~x86~~6.1.1.0
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Image Version: 6.1.7100.0
Package information:
Package Identity : Package_for_KB970419~31bf3856ad364e35~x86~~6.1.1.0
Applicable : Yes
Copyright : Microsoft Corporation
Company : Microsoft Corporation
Creation Time :
Description : Fix for KB970419
Install Client : WindowsUpdateAgent
Install Package Name : update.num
Install Time :
Last Update Time :
Name : default
Product Name : Package_for_KB970419
Product Version :
Release Type : Update
Restart Required : Possible
Support Information : http://support.microsoft.com/?kbid=970419
State : Not Present
Custom Properties:
<No custom properties found>
Features listing for package : Package_for_KB970419~31bf3856ad364e35~x86~~6.1.1.
0
<No features found for this package>
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>
```

**FIGURE 3-10** Information about an online package

You cannot remove an .msu file you have added to an image, but you can remove a .cab file by using the */remove-package* option. You can use the */get-packages* option to discover the package name as it is listed in the image and then use the */packagename* parameter in the command to identify it. Alternatively, you can use the */packagepath* parameter and specify the path to the source package you added; for example:

```
dism /image:c:\mountedimages /remove-package /packagepath:c:\mypackages\mypackage.cab
```



**EXAM TIP**

Remember that when you change an offline-mounted image by adding a package, removing a package, and so on you need to save the changes to the original source image by using the DISM */commit-wim* option.

The DISM options to manipulate and manage features are very similar to those that you use to work with packages. For example, to get information about the feature Chess in a running operating system, you would use the following command:

```
dism /online /get-featureinfo /featurename:Chess
```

If you want, you can use the */packagename* and */packagepath* parameters to find a specific feature in a package. You can enable or disable a specific feature in an image by using the */enable-feature* and */disable-feature* options, for example:

```
dism /image:c:\mountedimages /disable-feature /featurename:Minesweeper
```

## Package Installation Considerations

When you install a package in an offline image, the package state becomes “install pending,” and the package is installed when the image is booted and pending online actions are processed. If subsequent actions are requested, they cannot be processed until the previous pending online actions complete. If a package is in the “installed pending” state and you stage the package, the package state is set to “uninstall pending” because the package must be uninstalled before it can be staged.

Some packages require other packages to be installed first. If there are dependency requirements, you should use an answer file to install the necessary packages. By passing an answer file to DISM, you can install multiple packages in the correct order. Microsoft recommends the use of an answer file for installing multiple packages. Packages are installed in the order that they are listed in the command line, which in turn can be generated in an answer file.

When you use DISM to list the feature packages in a Windows PE image, the packages will always be listed as pending even when the servicing operation was successful. This is by design, and you do not need to take any further action.

## Configuring International Settings in an Image

You can use the DISM tool to manage international settings in a Windows 7 (or a Windows PE) image. You can also query existing settings in an offline or an online image.

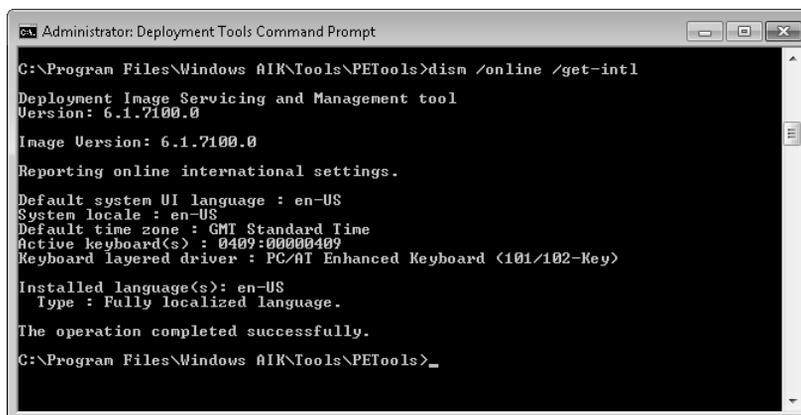
You can use the following international servicing options on an offline-mounted image:

```
dism /image:path_to_offline_image_directory [/get-intl] [/set-uilang |  
/set-uilangfallback | /set-syslocale | /Set-UserLocale | /Set-InputLocale | /Set-AllIntl  
| /Set-Timezone | /Set-SKUIntlDefaults | /Set-LayeredDriver] [/Gen-Langini |  
/Set-SetupUILang | /Distribution]
```

You can use the following command on a running operating system:

```
dism /online /get-intl
```

This is the only international servicing option you can apply to a running operating system. Its output is shown in Figure 3-11.



```
Administrator: Deployment Tools Command Prompt  
C:\Program Files\Windows AIK\Tools\PETools>dism /online /get-intl  
Deployment Image Servicing and Management tool  
Version: 6.1.7100.0  
Image Version: 6.1.7100.0  
Reporting online international settings.  
Default system UI language : en-US  
System locale : en-US  
Default time zone : GMT Standard Time  
Active keyboard(s) : 0409:00000409  
Keyboard layered driver : PC/AT Enhanced Keyboard <101/102-Key>  
Installed language(s): en-US  
Type : Fully localized language.  
The operation completed successfully.  
C:\Program Files\Windows AIK\Tools\PETools>_
```

FIGURE 3-11 International settings for an online operating system

If you use the `/get-intl` option with an offline-mounted image and specify the `/distribution` parameter, information about international settings and languages in the distribution share is displayed. Lesson 2 of this chapter discusses the distribution share.

If the language specified by the `/set-uilang` option is not installed in the Windows image, the command will fail. A fallback default language is required only when the language specified by the `/set-uilang` option is a partially localized language (for example, Ukrainian or Arabic).

The `/set-syslocale` option sets the language for non-Unicode programs (also called the system locale) and the font settings. You specify the name of the language and locale to set as the default language, for example, en-US. The `/set-userlocale` option configures a per-user setting that determines the default sort order and the default settings for formatting dates, times, currency, and numbers (for example, fr-FR).

### ✓ Quick Check

- You want more information about the package `Package_for_KB654321~22cf8952ad824e22~x86~~6.1.0.0` in a WIM image currently mounted in the folder `C:\MountedImages`. What command would you use?

### Quick Check Answer

- `dism /image:c:\mountedimages /get-packageinfo /packagename:Package_for_KB654321~22cf8952ad824e22~x86~~6.1.0.0`

The `/set-inputlocale` option sets the input locale and keyboard layout. For example, if you specify en-US as the local name, the option also sets the default keyboard layout defined for this locale. If you want to activate multiple keyboards in a single image, you can specify more than one keyboard layout by using semicolons as separators. The first value specifies the default keyboard. For example if you want to include the U.S. and U.K. keyboards in an image and use the U.K. layout as a default, you would enter a command similar to the following:

```
dism /image:c:\mountedimages /set-inputlocale:0409:00000409;0410:00010410
```

Figure 3-12 shows the output from this command.

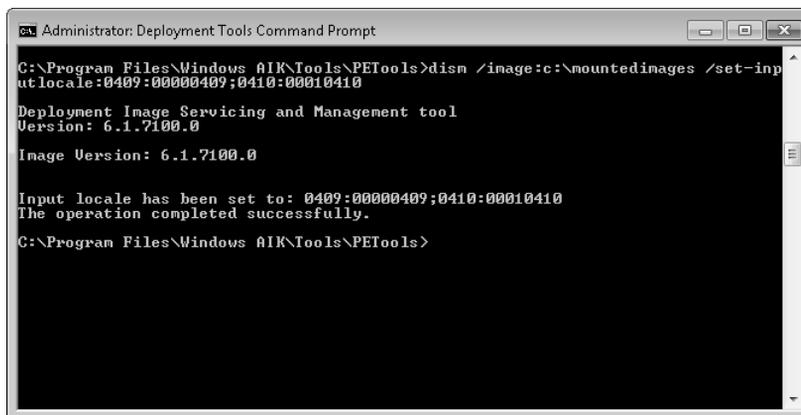


FIGURE 3-12 Specifying multiple keyboard layouts

If your image needs to include a keyboard driver for Japanese or Korean keyboards, you can use the `/set-layeredriver` option. This option takes an argument of 1 through 6 as follows:

1. PC/AT Enhanced Keyboard (101/102-Key)
2. Korean PC/AT 101-Key Compatible Keyboard/MS Natural Keyboard (Type 1)
3. Korean PC/AT 101-Key Compatible Keyboard/MS Natural Keyboard (Type 2)
4. Korean PC/AT 101-Key Compatible Keyboard/MS Natural Keyboard (Type 3)
5. Korean Keyboard (103/106 Key)
6. Japanese Keyboard (106/109 Key)

You can use the `/set-allintl` option to set the user interface (UI) language, system locale, user locale, and input locale to the same value, for example, en-US. If you use the `/set-allintl` option with any of the options that specify the individual language or locales, the individual settings take precedence.

You can also use the `/set-skuintldefaults` option to set an image's default system UI language, language for non-Unicode programs, standards and formats language, input locale, keyboard layout, and time zone values to the Windows 7 default value specified by a language name argument, such as en-US. Note that the `/set-skuintldefaults` option does not change the keyboard driver for Japanese and Korean keyboards. You use the `/set-layeredriver` option to specify this.

You can use the `/set-timezone` option to specify the default time zone. If you use this option, DISM verifies that the specified time-zone string is valid for the image. The name of the time zone must exactly match the name of the time zone settings in the registry in the `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\TimeZones\` registry key. If you add a custom time zone to your computer, you can specify that custom time-zone string.

#### **NOTE THE Tzutil COMMAND-LINE TOOL**

On a computer running Windows 7, you can use the `Tzutil` command-line tool to list the time zone for that computer. The `Tzutil` tool is installed by default on Windows 7 and is not part of the Windows AIK.

## Managing Windows Editions

Windows 7 edition packages for each potential target edition are staged within a Windows 7 `Install.wim` image on Windows 7 installation media. This is referred to as an edition-family image. Because the target editions are staged, you can service a single mounted image and apply the updates as appropriate to each edition in the image. This reduces the number of images you have to manage. However, it could increase the factory time or user time spent in the specialize configuration pass.

You can use the Windows edition-servicing commands to change one edition of Windows 7 to a higher edition within the same edition family. When you upgrade an offline image, you do

not require a product key. If you change an online image to a higher edition, you can add the product key using one of the following methods:

- Enter the product key during the out-of-box experience (OOBE) pass.
- Use an unattended answer file to enter the product key during the specialize configuration pass.
- Use DISM and the Windows edition-servicing command-line option `/set-productkey` after you configure the edition offline.

You can use the following edition-servicing options on an offline image to list editions or to change a Windows image to a higher edition:

```
dism /image:path_to_image_directory [/get-currentedition | /get-targeteditions |  
/set-edition | /set-productkey]
```

On a running Windows 7 operating system, the following edition-servicing options are available:

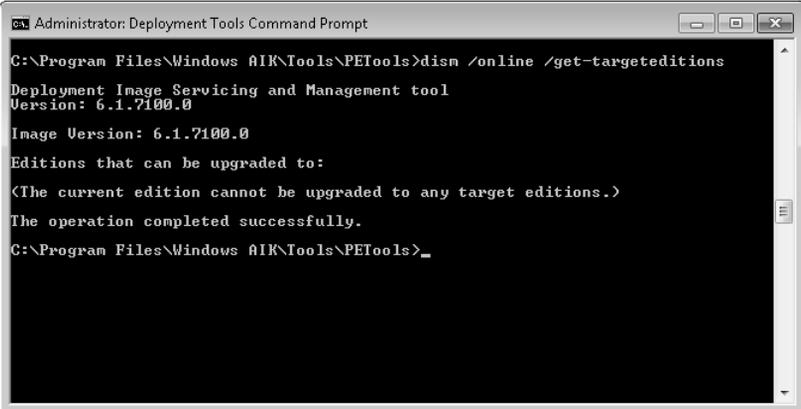
```
dism /online [/get-currentedition | /get-targeteditions]
```

Because this book is written for the Windows 7 installation image and product code on the installation media supplied, you will not be able to upgrade the mounted image you generated and placed on a bootable VHD and then subsequently mounted. The following commands entered on the online image or the installed image that you mount in the practice session exercises later in this lesson identify the online and mounted image Windows 7 editions:

```
dism /online /get-currentedition  
dism /image:c:\mountedimages / get-currentedition
```

Similarly you cannot upgrade your current edition to a target edition. As shown in Figure 3-13, the following command should tell you just that:

```
dism /online /get-targeteditions
```



```
Administrator: Deployment Tools Command Prompt  
C:\Program Files\Windows AIK\Tools\PETools>dism /online /get-targeteditions  
Deployment Image Servicing and Management tool  
Version: 6.1.7100.0  
Image Version: 6.1.7100.0  
Editions that can be upgraded to:  
(The current edition cannot be upgraded to any target editions.)  
The operation completed successfully.  
C:\Program Files\Windows AIK\Tools\PETools>_
```

**FIGURE 3-13** No target editions are available.

If a target edition is available, you can use the `/set-edition` option without the `/productkey` option to change an offline Windows image to a higher edition. Use `/get-targeteditions` to discover the edition ID. To change a running operating system to a higher edition, you can use the `/set-edition` option with the `/productkey` option, as in the following command:

```
dism /online /set-edition:Ultimate /productkey:12345-67890-12345-67890-12345
```

## Servicing Windows PE Images

You can mount a Windows PE image and add or remove packages, drivers, and language packs in the same way as you would for any other Windows 7 image. DISM also provides options specific to a Windows PE image. You can use these options to prepare the Windows PE environment, enable profiling, list packages, and prepare the Windows PE image for deployment.

For example, if you use DISM or ImageX to mount a Windows PE image in the folder `C:\MountedPEImage`, the options specific to Windows PE are as follows:

```
dism /image:c:\mountedpeimage [/get-pesettings | /get-profiling | /get-scratchspace |  
/get-targetpath | /set-scratchspace: | /set-targetpath : | /enable-profiling |  
/disable-profiling | /apply-profiles path_to_myprofile.txt]
```

### **NOTE** DISM WINDOWS PE OPTIONS APPLY ONLY TO OFFLINE-MOUNTED IMAGES

You cannot use DISM Windows PE options to manage an online, running version of Windows PE. You must specify a mounted Windows PE image using the `/image: path_to_image_directory` option.

You can obtain a list of PE settings in a mounted Windows PE image by entering a command similar to the following:

```
dism /image:c:\mountedpeimage /get-pesettings
```

You can discover whether the Windows PE profiling tool is enabled or disabled by entering a command similar to the following:

```
dism /image:c:\mountedpeimage /get-profiling
```

If you need to find out the amount of writeable space available on a Windows PE system volume when booted in RAMdisk mode, known as the Windows PE system volume scratch space, you can enter a command similar to the following:

```
dism /image:c:\mountedpeimage /get-scratchspace
```

Similarly, if you need to know the path to the root of the Windows PE image at boot time, known as the target path, you can enter a command similar to the following:

```
dism /image:c:\mountedpeimage /get-targetpath
```

You can set the scratch space and the target path by using commands similar to the following:

```
dism /image:c:\mountedpeimage /set-scratchspace:256
dism /image:c:\mountedpeimage /set-targetpath:D:\WinPEboot
```

Scratch space is specified in megabytes. Valid values are 32, 64, 128, 256, and 512. In hard disk boot scenarios, the target path defines the location of the Windows PE image on the disk. The path must be at least 3 characters and no longer than 32 characters. It must have a volume designation (C:\, D:\, and so on) and it must not contain any blank spaces.

File logging (or profiling) lets you create your own profiles in Windows PE 3.0 or later. By default, profiling is disabled. You can enable it, or disable it if previously enabled, by entering commands similar to the following:

```
dism /image:c:\mountedpeimage /enable-profiling
dism /image:c:\mountedpeimage /disable-profiling
```

When you create one or more profiles, each is stored in its own folder and identified in the file Profile.txt. You can remove any files from a Windows PE image that are not part of the custom profiles and check the custom profile against the core profile to ensure that custom application files and boot-critical files are not deleted, by entering a command similar to the following:

```
dism /image:c:\mountedpeimage /apply-profiles:c:\peprofiles\profile01\profile.txt,
c:\peprofiles\profile02\profile.txt
```

The paths to one or more profile.txt files are included in the command as a comma-separated list.

### ✓ Quick Check

1. You want to obtain a list of PE settings in a mounted Windows PE image in the folder C:\Mypeimage. What command do you enter in the elevated Deployment Tools command prompt?
2. You need to determine the amount of Windows PE system volume scratch space available on a Windows PE system volume in a mounted Windows PE image in the folder C:\Mypeimage when booted in RAMdisk mode. What command do you enter in the elevated Deployment Tools command prompt?

### Quick Check Answers

1. `dism /image:c:\mypeimage /get-pesettings`
2. `dism /image:c:\ mypeimage /get-targetpath`

## Unattended Servicing Command-Line Options

You can use DISM to apply an Unattend.xml answer file to an image. Typically, you would use this feature when you are installing multiple packages to the image. As stated previously in this lesson, some packages require other packages to be installed first. Microsoft recommends that the best way of ensuring the correct installation order is to use an answer file. If you use DISM to apply an Unattend.xml answer file to an image, the unattended settings in the offline Servicing configuration pass (previously described in Chapter 2) are applied to the Windows image.

The following servicing options are available to apply an Unattend.xml answer file to a offline Windows image:

```
dism /image:path_to_image_directory /apply-unattend:path_to_unattend.xml
```

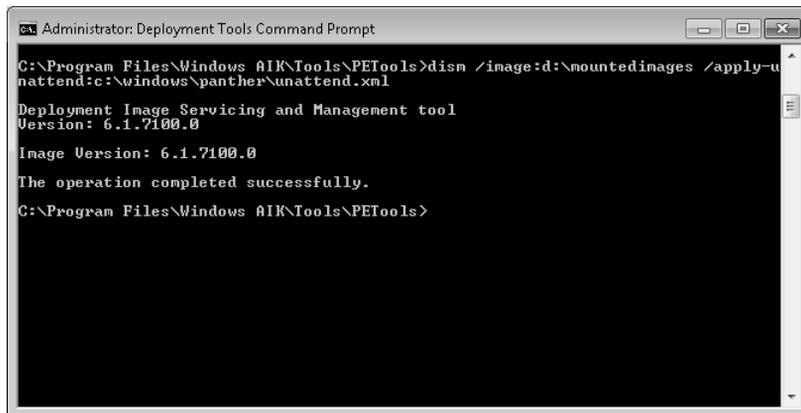
The following command applies an Unattend.xml answer file to a running operating system:

```
dism /online /apply-unattend:path_to_unattend.xml
```

For example, if the Unattend.xml file is located in C:\Windows\Panther, you can apply it to an offline-mounted image in C:\Mountedimages by entering the following command:

```
dism /image:c:\mountedimages /apply-unattend:c:\windows\panther\unattend.xml
```

Figure 3-14 shows the output from this command. It tells you the answer file has been applied but gives no additional information.



**FIGURE 3-14** Applying an answer file to an offline-mounted image

## Using Answer Files with Windows Images

The ability to associate an Unattend.xml answer file to an image provides a powerful tool to implement and configure image deployment, and to determine actions that can be taken if deployment fails or after deployment succeeds.

As discussed in Chapter 2, an answer file is an Extensible Markup Language (XML) file that contains setting definitions and values to use during Windows Setup. You specify Setup options in an answer file, including how to partition disks, the location of the Windows image to install, the product key to apply, and other custom Windows Setup settings. You can also specify values such as names of user accounts, display settings, and Windows Internet Explorer favorites. The answer file is typically called `Autounattend.xml` and is created using Windows System Image Manager (Windows SIM), as described in Chapter 2. If you want to add an additional answer file to install applications or specify the order in which packages are installed, you would typically use the file name `Unattend.xml`. When Windows SIM opens a Windows image file or catalog file, all of the configurable features and packages inside that image are displayed in the Windows Image pane. You can then add features and settings to the answer file.

### ✓ Quick Check

- Your offline-mounted WIM image file is in `C:\Images\Mounted`. An unattend answer file that you want to associate with this image has the file path `C:\Answerfiles\Unattend\Unattend.hml`. What command associates the answer file with the image?

### Quick Check Answer

- `dism /image:c:\images\mounted /apply-unattend:c:\answerfiles\unattend\unattend.hml`

Chapter 2 described how Windows SIM displays the properties and settings of a selected feature or package in its Properties pane. You can manage the feature settings for each configuration pass in this Properties pane. In the case of packages, editable Windows feature selections are displayed. Settings that are not available for each feature or package appear dimmed. Settings that have been edited appear in bold. Feature settings let you configure the aspects of each feature in a Windows 7 installation during unattended setup. For example, the Internet Explorer feature setting `Home_Page` can be configured to open to a particular URL by configuring the default value of the setting.

Feature properties are nonconfigurable attributes of the feature. Feature properties display differently when the feature is added to the currently open answer file. Feature IDs uniquely identify the feature of the operating system to which the settings belong. The ID contains the name, version, architecture, and so on for the feature selected in the Windows Image pane or Answer File pane. For example, the Language ID specifies the language code and the Name ID specifies the name of the feature or package. Package properties are nonconfigurable package attributes. For example, the ID Attribute package property specifies the identifier for the package in the following format: `ProcessorArchitecture_Version_Language_PublicKeyToken_VersionScope`.

Package settings are configurable attributes of a package placed in the Answer File pane, where you can edit them; for example, the Action package setting defines the action to be performed on the package. Possible actions are Install, Configure, Remove, or Stage.

Settings are sometimes organized into groups called list items. List items specify one or more values for a list item type. A list item type may include one or more feature settings. For example, you can create multiple favorites links by using the Favorite Item setting for Internet Explorer. Each list item must have a unique identifier, which is known as the key for that specific list item. If you use Windows SIM to manage list items, this enables you to add or delete a list item or modify its properties.

You can extend this concept to automate post-installation tasks. You can edit the answer file with a text editor in addition to Windows SIM. For example, if you are accustomed to writing batch files to automate application installation, you can add the same code to an answer file. Windows SIM creates a binary catalog file that lists all the settings in a Windows image.

You can create code manually or use Windows SIM to create a distribution that contains third-party drivers, applications, and Microsoft packages such as security bulletins. To create a distribution share, you must first create a distribution-share folder manually or by using Windows SIM. A *distribution share* is a shared Windows folder that contains the following subfolders:

- \$OEM\$ folders
- Packages
- Out-of-box drivers
- LangPacks

## Creating Answer Files

Microsoft recommends the use of Windows SIM to create unattend answer files, although you can also edit and create such files with a text editor such as Microsoft Notepad. If you use a manually authored answer file, you must validate the answer file in Windows SIM to verify that the answer file works. Answer files from Windows XP, Windows Server 2008, or Windows Vista do not work in Windows 7.

In general, it is best to expand to the lowest level of a feature and select only those elements that you intend to set. If you want to accept a default value, there is no need to include the element unless it is a required element. When creating answer files, you need to understand what happens during each configuration pass. Chapter 2 described configuration passes.

### **MORE INFO** CONFIGURATION PASSES

For more information about how configuration passes work, see [http://technet.microsoft.com/en-us/library/dd744341\(W5.10\).aspx](http://technet.microsoft.com/en-us/library/dd744341(W5.10).aspx).

When adding data, such as additional drivers or applications, take care that you do not overwrite Windows System files. Overwriting system files can corrupt your computer's operating system.

**MORE INFO** ADDING APPLICATIONS, DRIVERS, PACKAGES, FILES, AND FOLDERS

For more information about adding applications, drivers, packages, files, and folders, see [http://technet.microsoft.com/en-us/library/dd744568\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/dd744568(WS.10).aspx).

## Using Multiple Answer Files

You can use multiple second answer files (Unattend.xml) to create different custom images. For example, you could create a generic answer file that is used for each of your systems and then apply a second answer file during audit mode for changing disk configurations, drivers, or applications. To do this, you would use the Sysprep command (described in Chapter 2) with the `/unattend:answerfile` option. You can run this command manually during audit mode or you can add a custom command.

**MORE INFO** ADDING CUSTOM COMMANDS AND SCRIPTS

For more information about adding custom commands and scripts, see [http://technet.microsoft.com/en-us/library/dd744393\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/dd744393(WS.10).aspx).

## PRACTICE Mounting an Offline Image and Installing Language Packs

In this practice, you use both ImageX and DISM to mount an image. You also practice unmounting an image. You then apply language packs to a mounted image.

### EXERCISE 1 Mounting, Unmounting, and Remounting an Image

In this exercise, you use the ImageX tool to mount the system image Myimage.wim that you installed on the VHD to which you allocated drive letter W:. You mount the image in the folder C:\Mountedimages. You then unmount the folder. Finally, you use DISM to mount the image to the folder D:\Mountedimages. Note that it is not essential to create a different folder to hold the second mounted image. However, if you do not, it is a good idea to delete and re-create the original folder because DISM sometimes returns an error even though the image has been unmounted from the folder. Proceed as follows:

1. Log on to the Canberra computer with the Kim\_Akers account.
2. Create a folder called C:\MountedImages. If this folder already exists, ensure that it is empty.
3. On the Start menu, right-click Computer and choose Manage. Select Disk Management. If the W: disk does not appear in the Volume list, right-click Computer

Management and choose Attach VHD. Navigate to the Myvhd.vhd file in the C:\VHDs folder, as shown in Figure 3-15, and click OK. Drive W: should then appear. If necessary, close the AutoPlay dialog box.

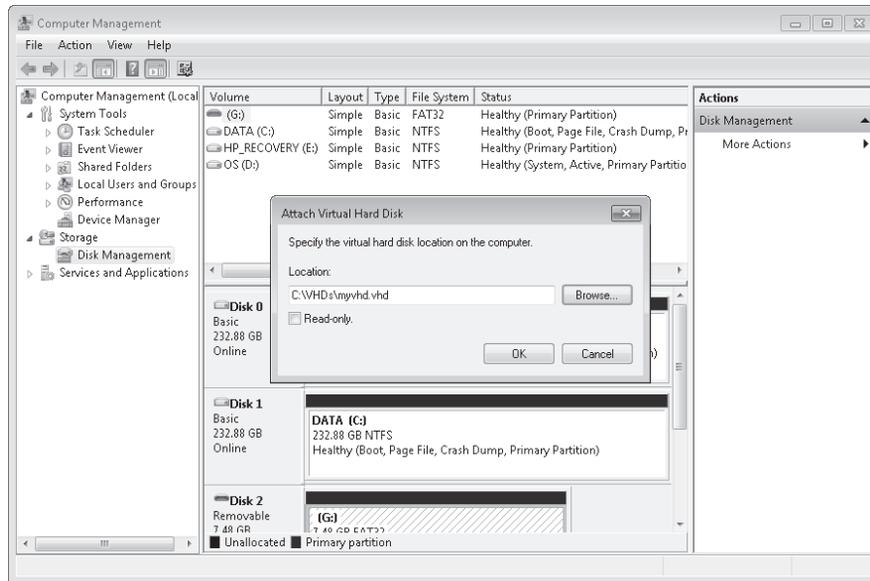


FIGURE 3-15 Attach the VHD if it does not appear in Computer Management.

4. On the Start menu, click All Programs, click Microsoft Windows AIK, right-click Deployment Tools Command Prompt, and choose Run As Administrator.
5. In the Deployment Tools command prompt, enter the command: **imagex /mountrw w:\myimage.wim 1 c:\mountedimages**. Figure 3-16 shows the output from this command.

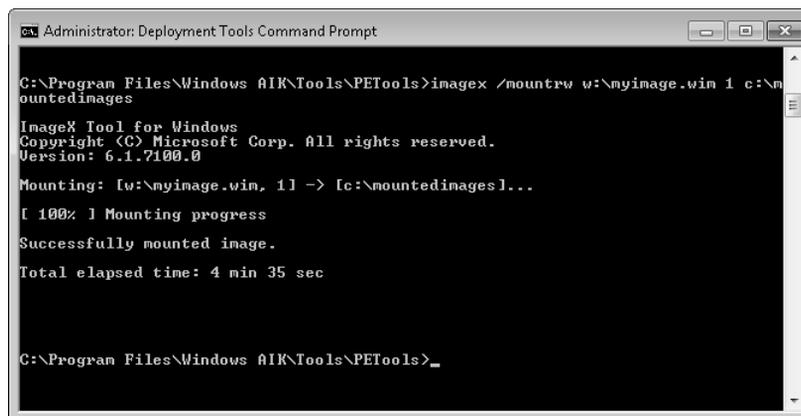
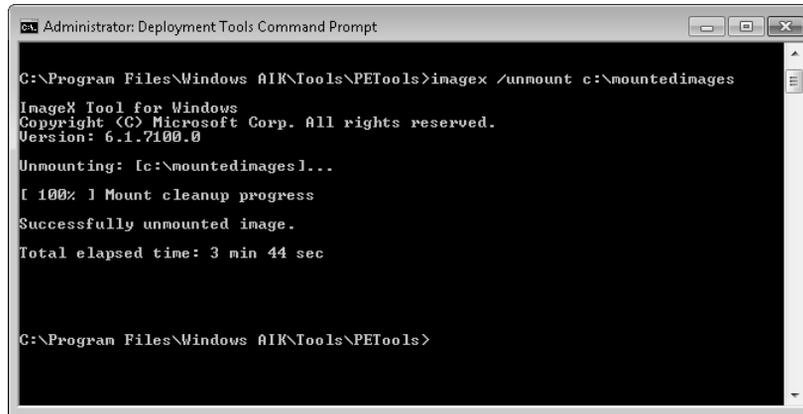


FIGURE 3-16 Using ImageX to mount a WIM image

- If you want, you can unmount the image and then use the DISM tool to mount it in a different folder. To unmount the tool, enter the command: **imagex /unmount c:\mountedimages**. Figure 3-17 shows the output from this command.

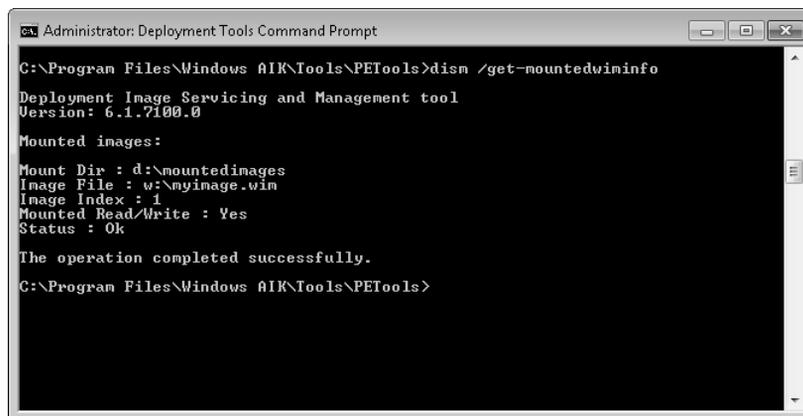


```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>imagex /unmount c:\mountedimages
ImageX Tool for Windows
Copyright (C) Microsoft Corp. All rights reserved.
Version: 6.1.7100.0
Unmounting: [c:\mountedimages]...
[ 100% ] Mount cleanup progress
Successfully unmounted image.
Total elapsed time: 3 min 44 sec

C:\Program Files\Windows AIK\Tools\PETools>
```

FIGURE 3-17 Unmounting an image

- Create a folder called D:\MountedImages. If you do not have a second hard disk, you can use the C:\MountedImages folder, but you might need to delete and re-create it if DISM returns an error.
- Mount the image with the DISM tool by entering the command: **dism /mount-wim /wimfile:w:\myimage.wim /index:1 /mountdir:d:\mymountedimages**.
- Test the image is mounted correctly by entering the command **dism /get-mountedwiminfo**. The output from this command is shown in Figure 3-18. Note that the mounted image folder is not the same as that shown in Figure 3-4 earlier in this lesson.



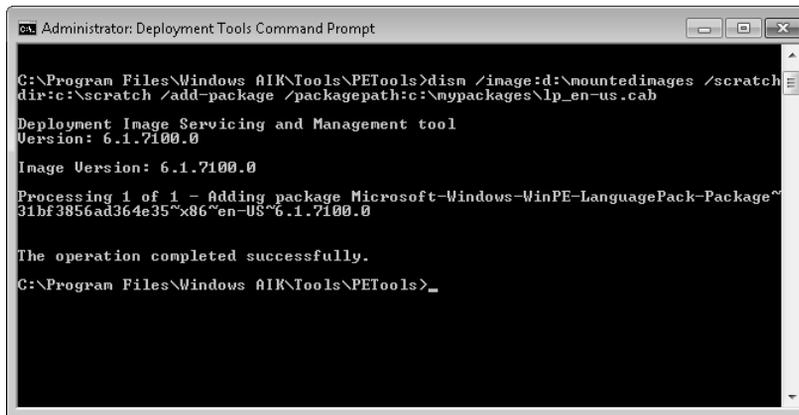
```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /get-mountedwiminfo
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Mounted images:
Mount Dir : d:\mountedimages
Image File : w:\myimage.wim
Image Index : 1
Mounted Read/Write : Yes
Status : Ok
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>
```

FIGURE 3-18 An image mounted in D:\Mountedimages

## EXERCISE 2 Applying a Language Pack to a Mounted Image

In this exercise, you apply the en-US language pack to your mounted image. You might do this if, for example, you already had an image with the lp\_fr-FR language pack installed and wanted to be able to configure international settings and distribute the image to both French-speaking and English-speaking areas. Note that you can apply multiple language packs only to Windows 7 Ultimate or Enterprise images. You should consider this a generic procedure because it is the way you would apply all packages that are distributed as cabinet (.cab) files. You need to have completed Exercise 1 before attempting this exercise. Proceed as follows:

1. If necessary, log on to the Canberra computer with the Kim\_Akers account.
2. If it does not already exist, create a folder called C:\Mypackages.
3. Navigate to C:\Program Files\Windows AIK\Tools\PETools\x86\WinPE\_FPs\en-us.
4. Copy the lp\_en-us cabinet file and save it in the C:\Mypackages folder.
5. Create a directory C:\Scratch. This will be used as the Scratch directory.
6. On the Start menu, click All Programs, click Microsoft Windows AIK, right-click Deployment Tools Command Prompt, and choose Run As Administrator.
7. Enter **dism /image:d:\mountedimages /scratchdir:c:\scratch /add-package /packagepath:c:\mypackages\lp\_en-us.cab**. Figure 3-19 shows the output from this command.



```
Administrator: Deployment Tools Command Prompt
C:\Program Files\Windows AIK\Tools\PETools>dism /image:d:\mountedimages /scratch
dir:c:\scratch /add-package /packagepath:c:\mypackages\lp_en-us.cab
Deployment Image Servicing and Management tool
Version: 6.1.7100.0
Image Version: 6.1.7100.0
Processing 1 of 1 - Adding package Microsoft-Windows-WinPE-LanguagePack-Package~
31bf3856ad364e35~x86~en-US~6.1.7100.0
The operation completed successfully.
C:\Program Files\Windows AIK\Tools\PETools>
```

FIGURE 3-19 Adding a language pack

8. To commit your changes to the source image, enter: **dism /commit-wim /mountdir:d:\mountedimages**. If you want, unmount the image.

## Lesson Summary

- You need to mount a writeable system image (WIM) file in a folder to service it. You can use the DISM or ImageX Windows AIK tools to mount an image.

- Very limited servicing options are available for an online running operating system, although you can use the DISM tool to discover information about the online image.
- You can use the DISM tool to add packages, drivers, and updates to a mounted image. You can obtain information about Windows Installer applications, application packages, and Windows features. You can disable and enable Windows features and display and configure international settings and Windows editions. You can apply unattended answer files to an image to implement hands-free installation and post-installation tasks. Finally, you can save the changes to the mounted image to the source image and unmount the image.
- You can use the DISM tool to mount and service Windows PE images.

## Lesson Review

You can use the following questions to test your knowledge of the information in Lesson 1, "Managing a System Image Before Deployment." The questions are also available on the companion DVD if you prefer to review them in electronic form.

### NOTE ANSWERS

Answers to these questions and explanations of why each answer choice is correct or incorrect are located in the "Answers" section at the end of the book.

1. You have copied the system image `Install.wim` file from your Windows 7 installation media to the folder `C:\Images`. You have mounted the image with index value 5 (Windows 7 Ultimate) to the folder `D:\Mount`. You want to add third-party drivers that you have stored in `C:\Drivers\Printer` and `C:\Drivers\Scanner` to the mounted image. Which of the following DISM commands would you use? (Choose all that apply.)
  - A. `dism /image:c:\images /add-driver /driver:c:\drivers /recurse`
  - B. `dism /image:d:\mount /add-driver /driver:c:\drivers /recurse`
  - C. `dism /image:c:\images /add-driver /driver:c:\drivers\printer /driver:c:\drivers\scanner`
  - D. `dism /image:d:\mount /add-driver /driver:c:\drivers\printer /driver:c:\drivers\scanner`
2. You need to find out the amount of writeable space available on a Windows PE system volume when booted in RAMdisk mode. The PE image is mounted in the folder `D:\PEMount`. What command would you use?
  - A. `dism /image:d:\pemount /get-scratchspace`
  - B. `dism /image: d:\pemount /get-targetpath`
  - C. `dism /image: d:\pemount /get-profiling`
  - D. `dism /image: d:\pemount /enable-profiling`

3. Which of the following DISM options can you run against an online, running operating system?
- A. `/set-syslocale`
  - B. `/set-userlocale`
  - C. `/set-inputlocale`
  - D. `/get-intl`
4. You have created an answer file called Unattend.xml in the C:\Textfiles\Answer folder. You want to apply it to an image mounted in the C:\Mount folder. What command would you use?
- A. `dism /image:c:\textfiles\answer /apply-unattend:c:\ mount \unattend.xml`
  - B. `dism /image:c:\mount /apply-unattend:c:\textfiles\answer\unattend.xml`
  - C. `dism /image:c:\mount /apply:c:\textfiles\answer\unattend.xml`
  - D. `dism /image:c:\mount /apply-answer:c:\textfiles\answer\unattend.xml`
5. You want to obtain detailed information about all the Windows Installer (.msi) applications installed in the WIM image mounted in the C:\Mount folder. What command do you use?
- A. `dism /online /get-packageinfo`
  - B. `dism /image:c:\mount /get-featureinfo`
  - C. `dism /image:c:\mount /get-appinfo`
  - D. `dism /image:c:\mount /get-apppatchinfo`

## Lesson 2: Deploying Images

Deploying images to large numbers of computers is a vital task in the enterprise environment. If 100 new client computers are purchased, you want to be able to deploy your current operating system, drivers, language packs, and so on with no errors and little or no user intervention. If a new user joins and a single client workstation is purchased, you want to be able to connect it to the network and have the appropriate image efficiently deployed.

Unfortunately, however, nothing stays the same, and few things go out of date more quickly than system images. A new driver is released, and a new edition of the software that your organization relies on arrives. New updates seem to appear daily (although they tend to show up mostly on Tuesdays). Some of them are important security updates and if you deploy your image without them, your clients are at risk. The company introduces additional hardware, and it is not Plug and Play.

This lesson looks at how you deploy images over a network, how you deal with image updates, and how you deploy to clients, some of which can boot automatically on to the network and some cannot. It discusses the tools you have available to perform this important administrative function efficiently.

### After this lesson, you will be able to:

- Use MDT to add updates, applications, and language packs to a disk image online and offline and keep WIM image files up to date.
- Create a deployment share to hold deployment images. Add deployment points and task sequences.
- Know the server tools such as WDS and SCCM 2007 that work with MDT 2010 or independently to deploy system images.
- Know the requirements for Lite Touch Installation (LTI) and Zero Touch Installation (ZTI).

**Estimated lesson time: 50 minutes**

## Using the Microsoft Deployment Toolkit

Chapter 2 briefly introduced the Microsoft Windows Deployment Toolkit (MDT) 2010. This toolkit is the Microsoft solution accelerator for operating system and application deployment and presents a number of new features, including flexible driver management, optimized transaction processing, and access to distribution shares from any location. In an enterprise environment, you would use the MDT on imaging and deployment servers to implement the automatic deployment of Windows 7 (for example) on client computers.

MDT 2010 unifies the tools and processes that you need for both desktop and server deployment into a deployment console. It features a fourth-generation deployment accelerator that integrates with Microsoft deployment technologies to create a single path

for image creation and automated installation. In other words, it makes the creation and deployment of a system image a lot easier. Microsoft states that MDT provides detailed guidance and job aids for every organized role involved with large-scale deployment projects. It offers unified tools and processes that you use for desktop and server deployment in a common deployment console and that reduce deployment time. The toolkit offers standardized desktop and server images, along with improved security and ongoing configuration management.

You can use MDT 2010 with the LTI method or it can be completely automated using ZTI. ZTI uses the Microsoft System Center Configuration Manager (SCCM) 2007 with the Operating System Deployment Feature Pack and also requires that a server running Microsoft SQL Server 2005 or SQL Server 2008 is available on the network. You can use LTI when software distribution tools are not in place to deploy to non-pre-execution environment (PXE)-compliant clients, although you need to use it with WDS to deploy to PXE-compliant clients. Although you install MDT 2010 on your Canberra computer in this lesson so you can investigate its features, it would be typically used on a deployment server with the WDS server role installed. Whatever deployment method you use, MDT 2010 requires that the Windows AIK is installed.

**NOTE SYSTEMS MANAGEMENT SERVER**

Unlike the previous MDT version (MDT 2008 Update 1), MDT 2010 cannot use Microsoft Systems Management Server (SMS) 2003 to implement ZTI.

Microsoft offers MDT 2010 in two versions to support Solution Accelerator feature installation on x64 or x86 hosts. The Quick Start Guide for Lite Touch Installation guide for MDT 2010 is available as a separate download for those who want to evaluate MDT 2010 quickly by viewing step-by-step instructions for using it to install Windows 7.

## New Features in MDT 2010

MDT 2010 offers a number of new features that are supported for LTI-based deployment. ZTI-based deployment using SCCM 2007 was introduced fairly recently (by MDT 2008 Update 1) and is mostly unaltered except that ZTI can no longer be implemented by using SMS 2003.

The MDT 2010 includes the following new features:

- Support for Windows 7
- Support for Windows Server 2008 R2
- Support for Windows AIK version 2.0
- Support for Windows User State Migration Toolkit (USMT) version 4.0. Specifically, the following new features of USMT 4.0 are supported in LTI-based deployments:
  - Support for USMT 4.0 hardlink migration
  - Support for USMT 4.0 shadow copy
- Support for the DISM tool

- Support for Windows PE version 3.0
- Support for the Boot Configuration Data (BCD) management tool and the BCDEdit command-line utility
- Support for Windows 7 default disk partition configuration. In MDT 2010, the disk partition configuration for Windows 7 places the operating system on Disk 0, Partition 2, and the system partition on Disk 0, Partition 1.

#### **MORE INFO** NEW MDT 2010 FEATURES

For more information about the new features MDT 2010 introduces, download the file [What's New in MDT 2010 Guide.docx](#), as described in the practice later in this lesson, or use Deployment Workbench to access the Information Center.

## MDT Program Folders

When you install MDT 2010, you create a number of subfolders in the MDT 2010 program folder %Sysvol%\Program Files\Microsoft Deployment Toolkit\ (typically C:\Program Files\Microsoft Deployment Toolkit\). Table 3-3 describes these subfolders.

**TABLE 3-3** MDT 2010 Program Folders

SUBFOLDER	DESCRIPTION
Bin	Holds the Deployment Workbench MMC snap-in and supporting files.
Control	Holds configuration data for Deployment Workbench. Typically, this folder is empty directly after installation.
Documentation	Holds documentation and job aids such as a splash screen for MDT 2010.
Downloads	Holds a feature list for features that Deployment Workbench downloads.
Management Pack	Holds management pack files, for example Microsoft.Deployment.Management.Pack.xml.
Samples	Holds sample task sequence scripts (for example, ZTICache.vbs) and Windows PE desktop background graphics.
SCCM	Holds task sequence templates and automation objects used during SCCM integration; for example, Deploy_SCCM_Scripts.vbs.
Scripts	Holds scripts that Deployment Workbench uses; for example ComponentCheck_scripts.vbs.
Templates	Holds template files that Deployment Workbench uses.

## Using Deployment Workbench

When you have installed MDT 2010, you can start Deployment Workbench from the Microsoft Deployment Toolkit program suite. You will be using this tool extensively in this lesson to deploy a Windows 7 system image. This section gives an overview of the features the tool offers. Deployment Workbench gives you access to the following items:

- **Information Center** This lets you access MDT 2010 documentation, including the latest news about MDT 2010 and the features you require to use it.
- **Distribution Share** This gives you a checklist of tasks you need to perform to deploy an operating system image, as shown in Figure 3-20. You also use this tool to create a distribution directory, which is the second task on the list. You installed the Windows AIK in Chapter 2.

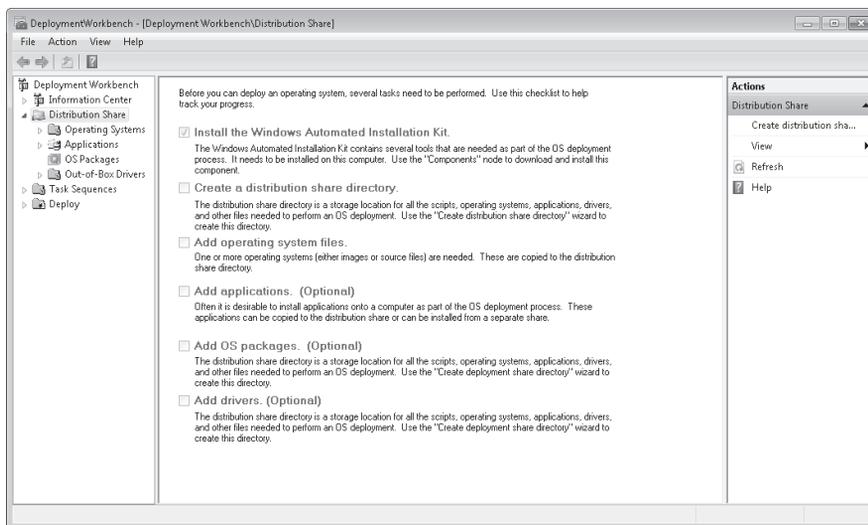


FIGURE 3-20 Task checklist

- **Task Sequences** This provides a list of task sequences in the details pane. To create a task sequence, right-click Task Sequences and then click New. To configure a task sequence, right-click it in the details pane and then click Properties.
- **Deploy** You can expand this item to see the Deployment Points and Database items. Click Deployment Points to see a list of deployment points in the details pane. To create a deployment point, right-click Deployment Points, and then click New. To configure a deployment point, right-click a deployment point in the details pane, and then click Properties. Click Database to edit the database.

### **CAUTION** OPEN ONLY A SINGLE INSTANCE OF DEPLOYMENT WORKBENCH

Microsoft recommends that you open only a single instance of Deployment Workbench. Opening two or more instances can result in unpredictable behavior.

## Choosing an Image Strategy

If you are distributing an image across an enterprise environment, your aim should be to create a standard configuration that is based on a common image for each version of an operating system. Organizations want to apply a common image to any computer in any region at any time, and then customize that image quickly to provide services to users.

Most organizations build and maintain many images. However, you can reduce the number of different images by making disciplined hardware purchases and by using advanced scripting techniques. You can utilize the software distribution infrastructure necessary to deploy applications and to keep your images updated.

You can use one of the following image types depending on whether you want to install only operating systems to large numbers of computers, whether you want to deploy applications, language packs, and other files at the same time as operating systems, or whether you are deploying an image to a VHD on a single computer for backup and failover purposes:

- Thick image
- Thin image
- Hybrid image

Thick images contain core applications, language packs, and other files in addition to the operating system. When you create a disk image that contains core applications and language packs, you need only a single step to deploy the disk image and core applications to the target computer, with language support for all target locales. Also, thick images can be less costly to develop, because they frequently do not require advanced scripting technique. You can use MDT 2010 to build thick images with little or no scripting. If you use thick images, core applications and language packs are available on first start.

The disadvantage of thick images is that updating a thick image with a new version of an application or language pack requires rebuilding, retesting, and redistributing the image. If thick images are built that include core applications and language packs, you need to install the core applications and language packs during the disk imaging process. You use thick images when you employ WIM files for backup and failover on bootable VHDs on individual computers running Windows 7 Enterprise or Ultimate.

A thin image carries a much lower cost to maintain and store. It contains few (if any) core applications or language packs. Applications and language packs are installed separately from the disk image. If you need to mitigate the network transfer time, you can use trickle-down technology such as Background Intelligent Transfer Service (BITS). Many software distribution infrastructures provide this facility.

The main disadvantage of thin images is that they can be more complex to develop. Deploying applications and language packs outside the disk image requires scripting and a software distribution infrastructure. If you use thin images, core applications and language packs are not available on first start. In some scenarios, this is regarded as a security risk.

If you choose to build thin images that do not include applications or language packs, your organization should have a systems management infrastructure such as SCCM 2007 in place to deploy applications and language packs. You should use this infrastructure to deploy applications and language packs after installing the thin image.

Hybrid images mix thin- and thick-image strategies. In a hybrid image, the disk image is configured to install applications and language packs on first run but automatically installs the applications and language packs from a network source. Hybrid images present most of the advantages of thin images, but they are not complex to develop and do not require a software distribution infrastructure. They do, however, require longer installation times.

You can choose to build a one-off thick image from a thin image by building a reference thin image. Then, you can add core applications and language packs, capture them, test them, and distribute a thick image based on the thin image. However, be wary of applications that are not compatible with the disk imaging process.

Hybrid images store applications and language packs on the network but include the commands to install them when you deploy the disk image. This process differs from installing the applications and language packs in the disk image because the image deployment process installations that would typically occur during the disk imaging process is deferred.

## Managing and Distributing Images with MDT 2010

When you create an image on a distribution share using the MDT 2010 utility, you often need to add updates, language packs, and applications. You can do this both offline and online with the MDT 2010 tool.

Although you are unlikely to use a computer running a Windows 7 client operating system as a distribution server in an enterprise environment, you nevertheless can install the MDT 2010 tool and use it to create a deployment share, deployment points, and task sequences. You can use MDT 2010 running on a machine running Windows 7 on a small test network to install client operating systems. You might possibly also do this in a small workgroup environment, although Microsoft recommends the use of a server running Windows Server 2008 Foundation for this purpose.

MDT 2010 is a mechanism for managing and distributing WIM images. If you want to configure a Windows 7 operating system on a large number of client computers, you can obtain a WIM image (Install.wim) from the installation media. Depending on the type of computers in the environment and the hardware they contain, you may need to add device drivers from hardware vendors to make the system fully functional. You also need to add updates to the image.

You can distribute the operating system image to client computers using either the LTI or the ZTI method. The latter method requires that you have SCCM 2007 along with SQL Server 2005 or SQL Server 2008 available on your network.

As described in Chapter 2, you can install Windows 7 on a reference computer, together with any software you want to include as part of the image, and use the Sysprep tool to prepare the computer for imaging. You can then boot into Windows PE and use the ImageX Windows AIK tool to generate a WIM image file that you can place in the MDT distribution share.

**NOTE TWO TYPES OF REFERENCE COMPUTER**

If you are using a thin-image approach and using Deployment Workbench to add updates, drivers, packages, language packs, and applications to an operating system image copied from installation media, you should always deploy the resulting image to a single client computer and test it thoroughly before deploying to a number of clients. The single client computer used for testing is often termed the *reference computer*. Distinguish between a reference computer that you install and configure manually and test before you capture its image (as described in Chapter 2) and a reference computer that you install from an image that you have configured in MDT 2010 for testing purposes.

**✓ Quick Check**

- You have installed Windows AIK and MDT 2010. What additional software tools do you require to implement ZTI?

**Quick Check Answer**

- SCCM 2007 and SQL Server.

## Creating a Distribution Share

Your first step in using MDT 2010 to deploy a system image is to create a distribution share to hold that image. You do this in the practice later in this lesson, but a high-level procedure is described here. The distribution share contains all the information and settings that MDT 2010 uses. To create a new distribution share, carry out the following procedure:

1. Open Deployment Workbench, right-click Distribution Share, and click Create Distribution Share Directory.

2. In the Specify Directory page of the Create Distribution Share Wizard, click Create A New Distribution Share.
3. Type the location for the distribution share on the local system in the Path For New Distribution Share Directory text box, and then click Finish.

You can use Deployment Workbench to configure the distribution share to implement the following tasks:

- Add, remove, and configure operating systems.
- Add, remove, and configure applications.
- Add, remove, and configure operating system packages, including updates and language packs.
- Add, remove, and configure out-of-box device drivers.

The source files for these tasks are stored in the distribution share folder and are associated with task sequences during the configuration process. Deployment Workbench stores metadata about operating systems, applications, operating system packages, and out-of-box device drivers in the distribution share's Control subfolder.

## Adding an Operating System Image

If you have created a custom image either by imaging a reference computer, as described in Chapter 2, or by imaging a client workstation running Windows 7 Enterprise or Ultimate to place that image on bootable VHD for failover, you can add that operating system image to the distribution share on a computer running MDT 2010. This is typically a thick-image approach because your reference image will contain updates, applications, drivers, and packages (including language packs). You can also add WDS images from WDS servers to the distribution share.

However, possibly the most typical scenario is that you add an image and all its associated installation files from installation media. You do this in the practice later in this lesson. Typically, this gives you flexibility because a single Install.wim file can contain images for several Windows 7 editions. It is a thin-image approach because images on the installation media will not include any third-party drivers, Windows Installer files, mission-critical applications, or additional language packs. Most significantly, critical security packs issued since the installation image was created will not be present. Applying such images to multiple client computers and bringing these computers online represents a security risk unless at least critical security updates are added to the image before deployment.

To deploy images using MDT 2010, you need to specify the source directory in which the WIM file resides. The New OS Wizard moves the file to the distribution share. To add an operating system image to the distribution share, perform the following high-level procedure:

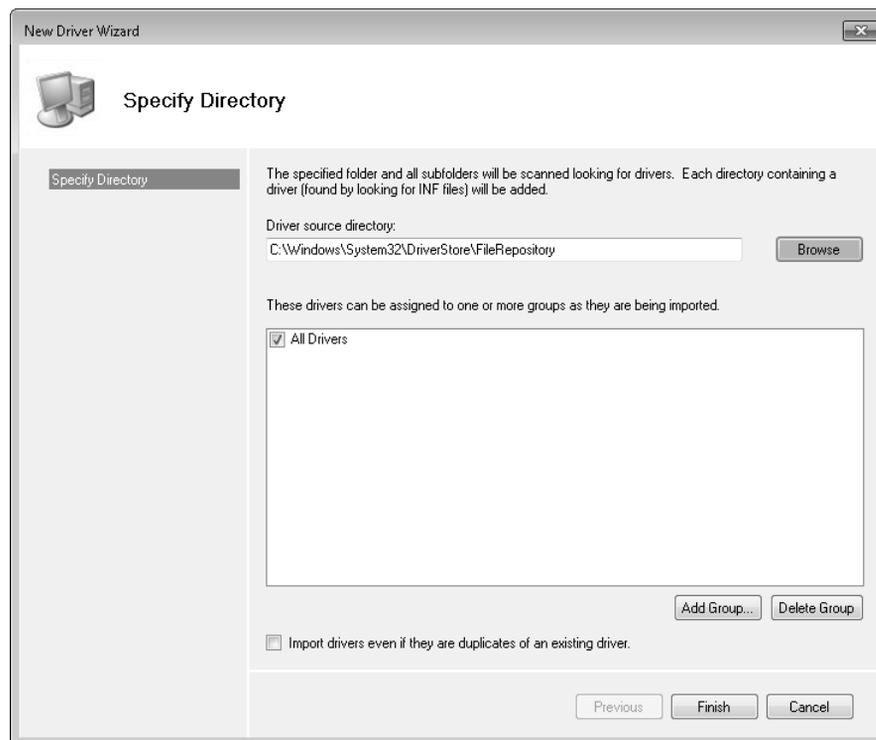
1. In the Deployment Workbench console tree, expand Distribution Share, right-click Operating Systems, and click New (or click Operating Systems and click New in the Actions Pane). This starts the New OS Wizard.

2. On the OS Type page, select the type of image (custom, installation files, or WDS) that you want to add.
3. Select an image source and then specify a target folder that is a subfolder of the distribution share. In the case of custom images, you have the option of moving the files to the distribution share instead of copying them.

## Adding Device Drivers

After the operating system has been added to Deployment Workbench, add any required device drivers. During operating system deployment, these drivers are added to Windows PE and the target computers and deployed with Windows 7. The New Driver Wizard in Deployment Workbench copies the device driver files to the distribution share. The high-level procedure for adding out-of-box (third-party) device drivers is as follows:

1. In Deployment Workbench, expand Distribution Share, click Out Of Box Drivers, and click New in the Actions pane.
2. In the New Driver Wizard, browse or type the path to the folder in which the drivers you want to add are stored. Figure 3-21 shows a path to a folder that holds drivers on the Canberra computer, but in practice, you would create a folder to hold specific third-party drivers that you want to install.



**FIGURE 3-21** Specifying a path to driver files

3. Click Add Group and add a driver group, for example, Windows 7. Take a note of the name of this group—you need this information to create bootable LTI media.
4. Select the check box for the new driver group.
5. If you want, you can select Import Drivers Even If They Are Duplicates Of An Existing Driver.
6. Click Install.

The out-of-box device drivers install in subfolders of the C:\Distribution\Out-Of-Box Drivers folder.

## Configuring Task Sequences

You use task sequences to add updates, language packs and other packages, and applications to an image. You also specify a task sequence that binds operating system source files with a configuration. The system source files include the following:

- **Operating system** You can choose an operating system or custom image to use for the build.
- **Unattended Setup answer file (Unattend.xml or Autounattend.xml)** You can create an answer file that describes how to install and configure Windows 7 on the target computer. For example, the answer file can contain a product key, organization name, and information necessary to join the computer to a domain. Chapter 2 described how you use Windows SIM to create unattended answer files.
- **Task sequence** Each build has a default task sequence for hands-free installation.

## Creating and Editing a Task Sequence

The Task Sequencer runs the task sequence in the order specified. Each task in the sequence is a step, and steps can be organized into groups and subgroups. When creating a task sequence in Deployment Workbench, the tool creates a default task sequence. You can edit tasks and groups. You can also import task sequences created by other software packages, for example, SCCM 2007.

Task sequences contain the following item types:

- **Tasks** Within a task sequence, tasks are commands that the Task Sequencer runs during the sequence, such as partitioning the disk, capturing user state, and installing the operating system. In the default task sequence, most tasks are commands that run scripts.
- **Groups** The task sequence can be organized into groups, which are folders that contain subgroups and tasks. For example, the default task sequence puts tasks in groups by phase and deployment type. Tasks and groups (including the groups and tasks they contain) can be filtered based on specified conditions. Groups are useful for filtering because an entire collection of tasks can be run based upon a condition such as the deployment phase or type of deployment.

You create task sequences by using the Task Sequence Editor in Deployment Workbench. Each sequence consists of a series of steps designed to complete a specific task. Task sequences can operate across a computer restart and typically are configured to automate tasks on a computer without user intervention. Task sequence steps can be added to a task sequence group, which help keep similar task sequence steps together for better organization and error control.

Task sequence steps can use utilities and scripts provided with MDT 2010 or written as custom solutions for a particular task. You use the Task Sequence Editor to specify the task sequence groups, task sequence steps, and the valid properties and options to use to configure each part of the image preparation and deployment process. You need to provide the following information for each task sequence group and step:

- **Name** Names the task sequence group or step
- **Description** Describes the purpose of the task sequence group or step and provides information regarding its customization
- **Properties** Specifies the configuration properties for the task sequence group or step and defines how the task is performed
- **Options** Indicates the configuration options that can be specified for the task sequence group or step, when the task is performed, and successful exit conditions

**NOTE** ADDITIONAL TASK SEQUENCE STEP TYPES

Additional task sequence step types and conditional statements typically are available if you configure task sequences using SCCM 2007.

The high-level procedure to create a task sequence using MDT 2010 is as follows:

1. In Deployment Workbench, select Task Sequences and then click New in the Actions pane. In the General Settings page of the New Task Sequence Wizard, specify the Task Sequence ID and Task Sequence name. If you want, add comments.
2. On the Select Template page, select a task sequence template from the list shown in Figure 3-22.
3. On the Select OS page, select an operating system to be associated with the task sequence. Your choice is limited to the operating systems contained in the WIM image that you added to the deployment share.
4. You can specify the product key on the Specify Product Key page, or you can choose to provide this information at the time of deployment.
5. On the OS Settings page, provide your full name, your organization, and your organization's Internet Explorer home page.
6. At this point, you can choose to supply and confirm an administrator password on the Admin Password page or to supply this information later at deployment time.
7. Click Finish to create the task sequence. The task sequence appears in Deployment Workbench, as shown in Figure 3-23.

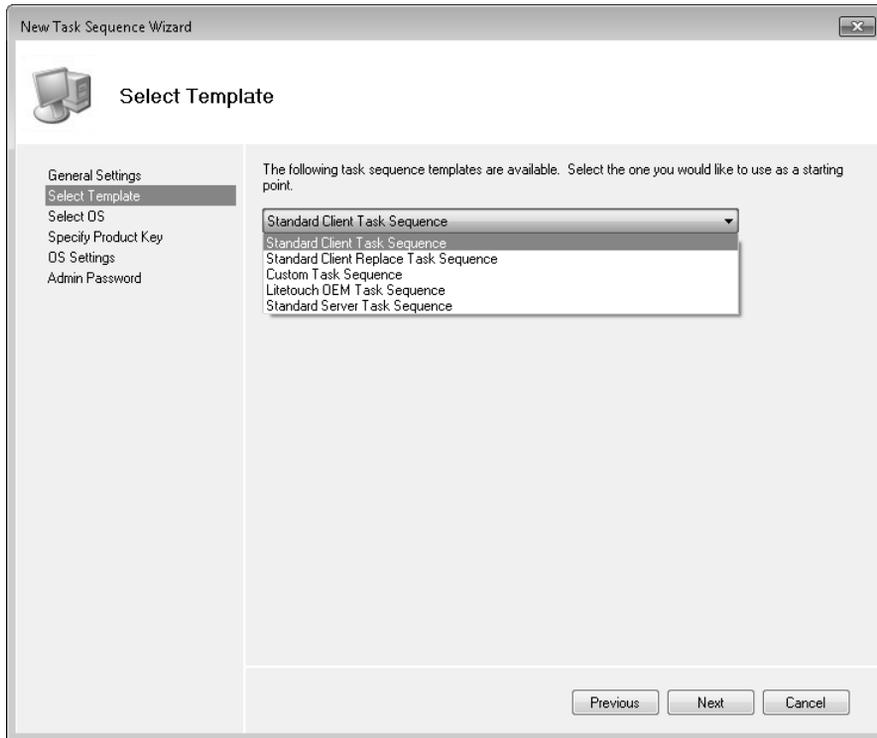


FIGURE 3-22 Available templates

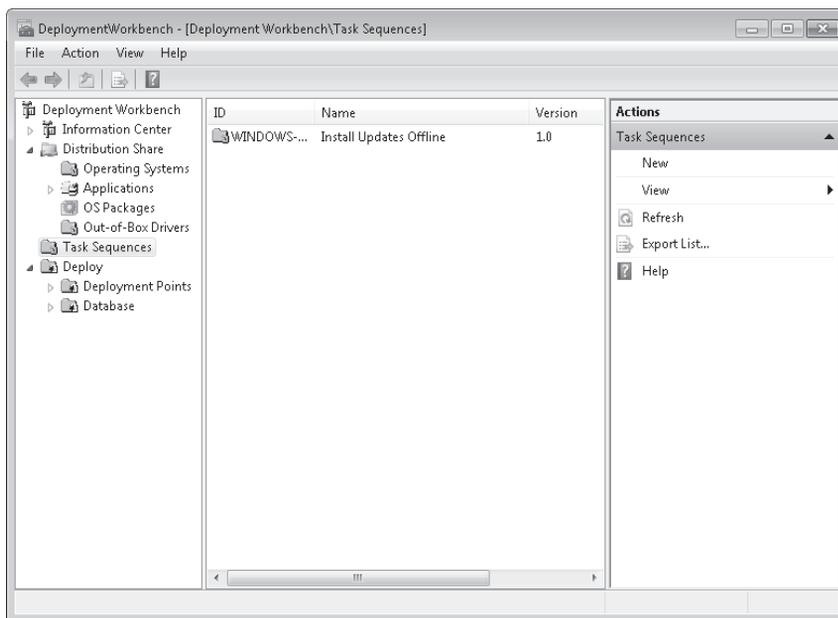
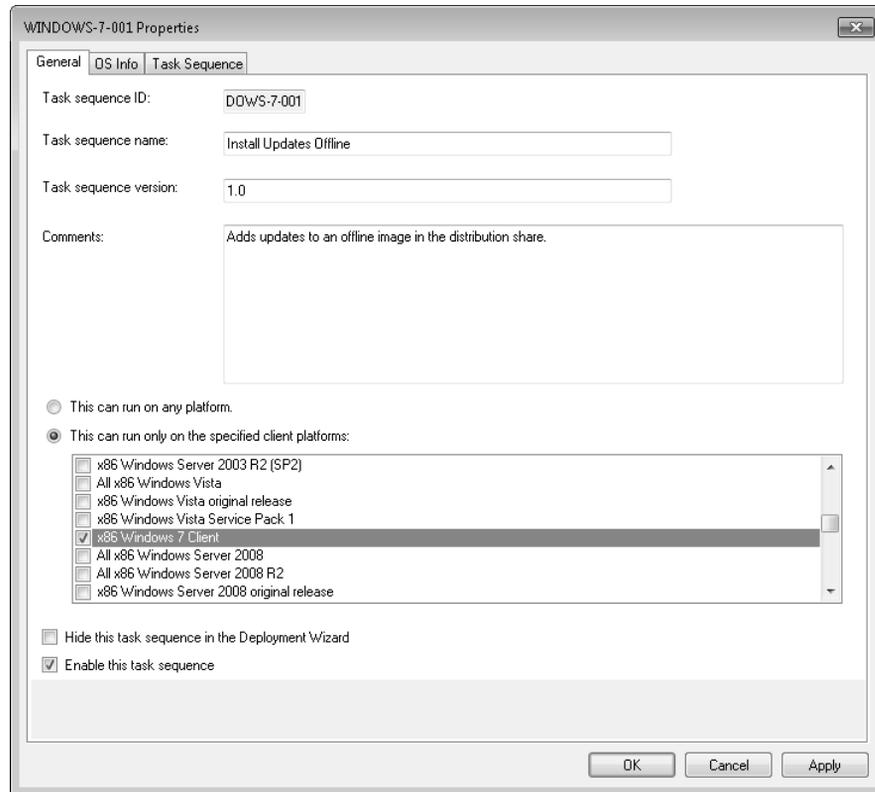


FIGURE 3-23 A task sequence in Deployment Workbench

- To edit the task sequence, right-click it and choose Properties. On the General tab, you can change settings such as the sequence name. You can also specify to which client operating system the task sequence can be applied, as shown in Figure 3-24.

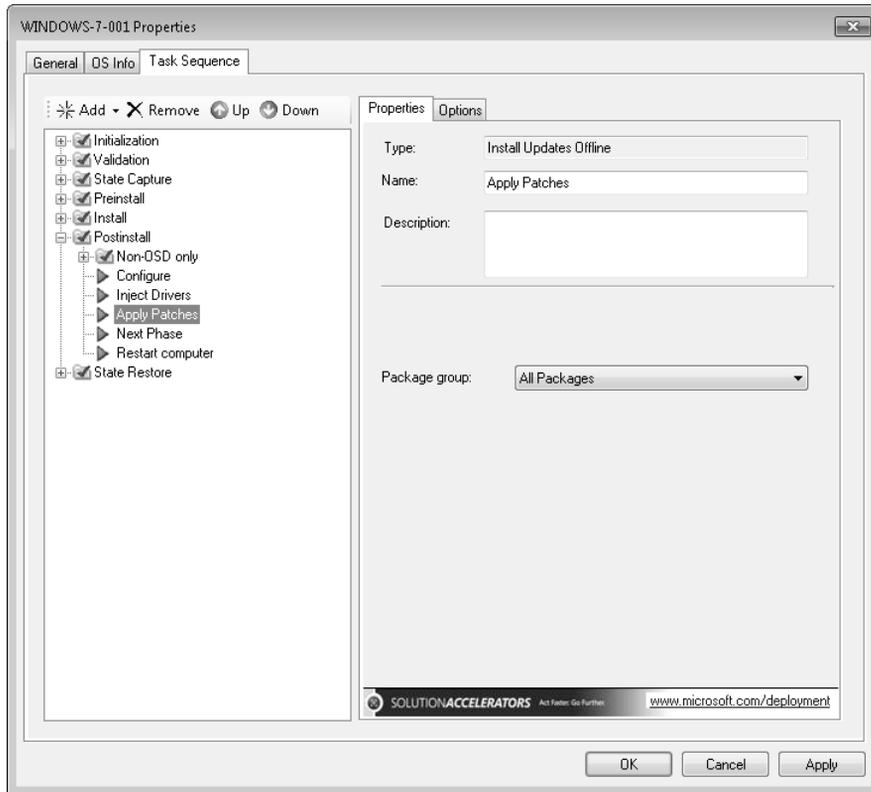


**FIGURE 3-24** Editing task sequence properties on the General tab

- The Task Sequence tab for a new task sequence shows the template chosen for that task sequence. Figure 3-25 shows the Standard Client Task Sequence template. You can edit this (typically by deleting unwanted tasks) so your new task sequence carries out the required tasks. The Options tab on the Task Sequence tab lets you disable the step, set a Continue On Error condition, or, in some cases, specify a Success Code.

#### **MORE INFO** TASK SEQUENCE EDITOR

For more information about the Task Sequence Editor, see *Operating System Deployment: Task Sequence Editor* at <http://technet.microsoft.com/en-us/library/bb680396.aspx>.



**FIGURE 3-25** The Standard Client Task Sequence template

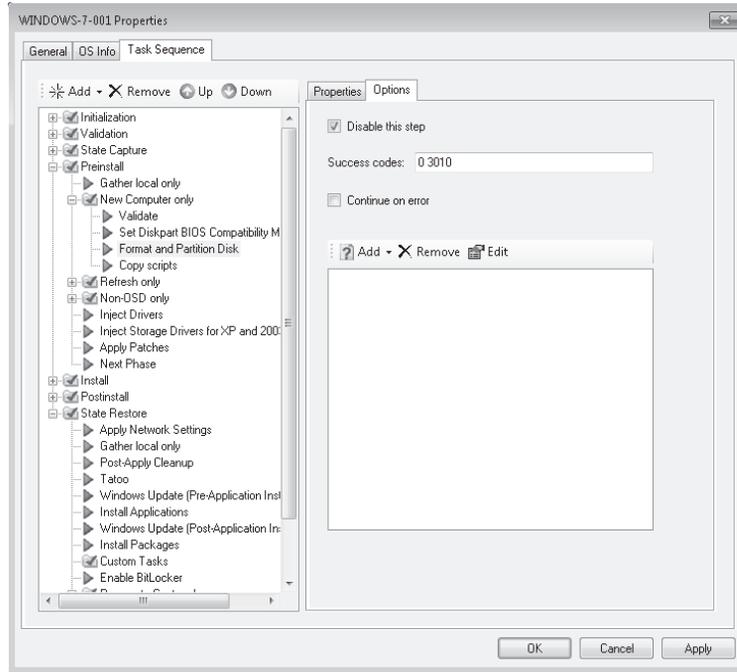
## Creating a Task Sequence to Deploy Windows 7 to VHD

Sometimes you might want to deploy a Windows 7 image to boot from VHD on your client computers. You might have users whose computers currently run Windows Vista and who want to have the option of a dual boot so they can try Windows 7 before committing to it. One option is to create virtual machines running Windows 7 on the client computers. Doing this through centralized deployment, however, is not straightforward.

A second alternative is to use MDT 2010 to deploy Windows 7 to a bootable VHD on each client computer. Users can then choose to boot into Windows Vista or boot from VHD into Windows 7. This is an attractive option for client computers that have only a single hard disk because it avoids having to repartition the drive for Windows 7 while keeping the original operating systems intact.

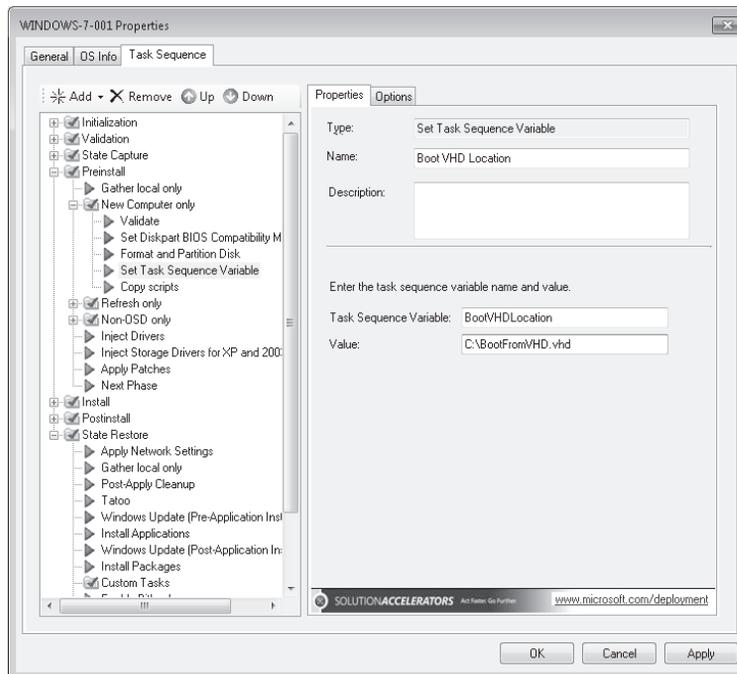
By default, MDT 2010 deploys Windows 7 to Disk 0 Partition 1. To change this behavior so that MDT 2010 deploys to a VHD and leaves the existing operating system intact, you need to create a standard Task Sequence that installs Windows 7 and then modify it as follows:

1. In the Task Sequence tab of the Task Sequence Properties box, expand Preinstall, Expand New Computer Only, and click Format And Partition Disk.
2. Disable this step, as shown in Figure 3-26.



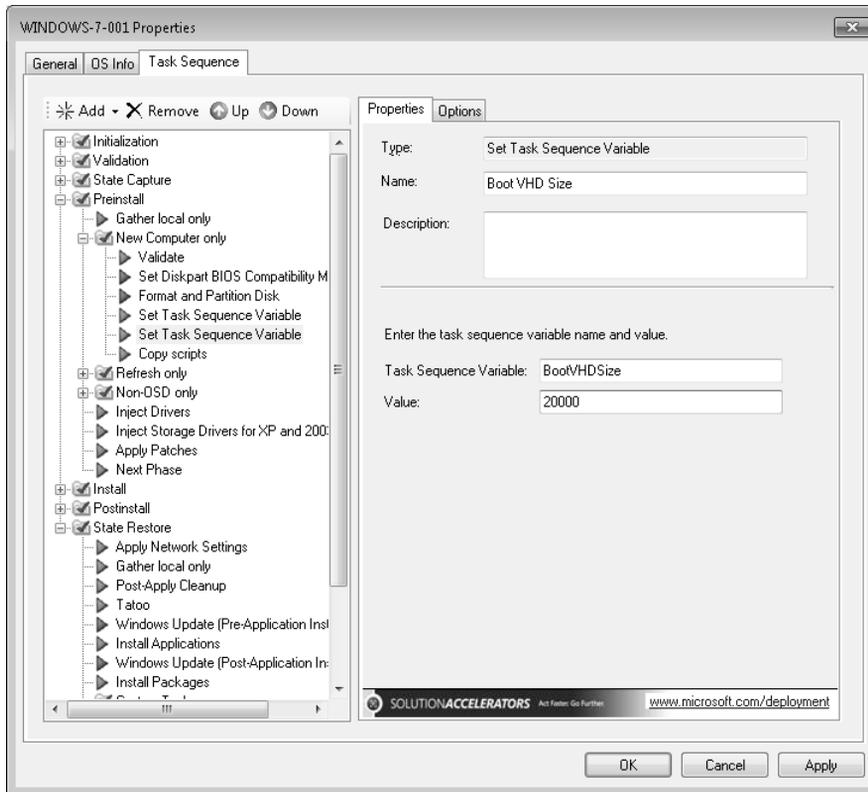
**FIGURE 3-26** Disabling the Format And Partition Disk step

3. Click Add, click General, and click Set Task Sequence Variable. Configure the Task Sequence Variable BootVHDLocation, as shown in Figure 3-27.



**FIGURE 3-27** Configuring the BootVHDLocation variable

4. Click Add, click General, and click Set Run Command Live. Configure the Task Sequence Variable `BootVHDSize`, as shown in Figure 3-28. Ensure that there is enough space on the drive (20 GB) because the VHD may grow to this size.



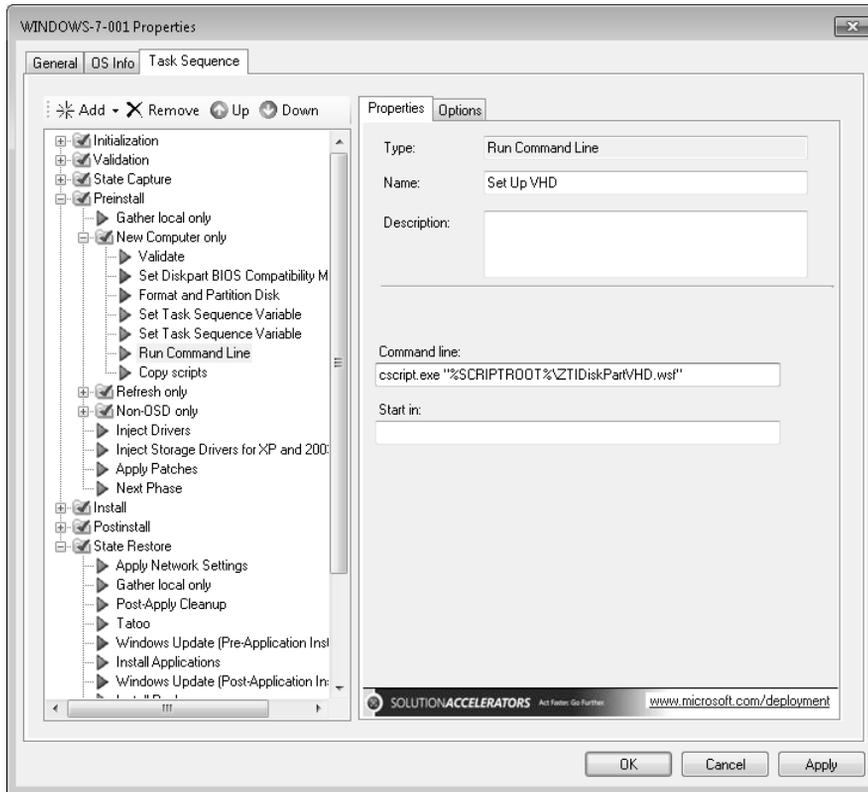
**FIGURE 3-28** Configuring the size of the VHD

5. Click Add, click General, and click Set Task Sequence Variable. Name the variable Set Up VHD. Insert the Run Command Line step `cscript.exe "%SCRIPTROOT%\ZTIDiskPartVHD.wsf"`, as shown in Figure 3-29.
6. Click Apply.

Using this Task Sequence, Windows 7 is deployed to a VHD rather than to an internal hard disk.

## Adding Updates

When developing an image, you should ensure that all critical security updates are included in the image. You can use different approaches to perform these updates. Microsoft recommends that if possible, you should add updates offline. The following options are available (provided a distribution share exists):



**FIGURE 3-29** Inserting a Run command

- **Add updates offline** You can use MDT 2010 to install updates offline using a task sequence. In the Deployment Workbench Task Sequence Editor, select the Install Updates Offline task. This option is available only if SCCM 2007 is available on your network.
- **Add updates online** You can use MDT 2010 to install updates online using a task sequence. In the Deployment Workbench Task Sequence Editor, select the Install Updates Online task. As with the previous option, SCCM 2007 needs to be available on your network.
- **Add updates to the master image** You can download security updates from the Microsoft Update Web site, and then install them as part of the image build process if you are applying a custom image. Additional updates can be added by placing the downloaded updates in the distribution share. You can add updates to an installation image copied from installation media by downloading them from the Microsoft Update Catalog at <http://catalog.update.microsoft.com/v7/site/Home.aspx> and adding them to a distribution share. Figure 3-30 shows the Microsoft Update Catalog home page.

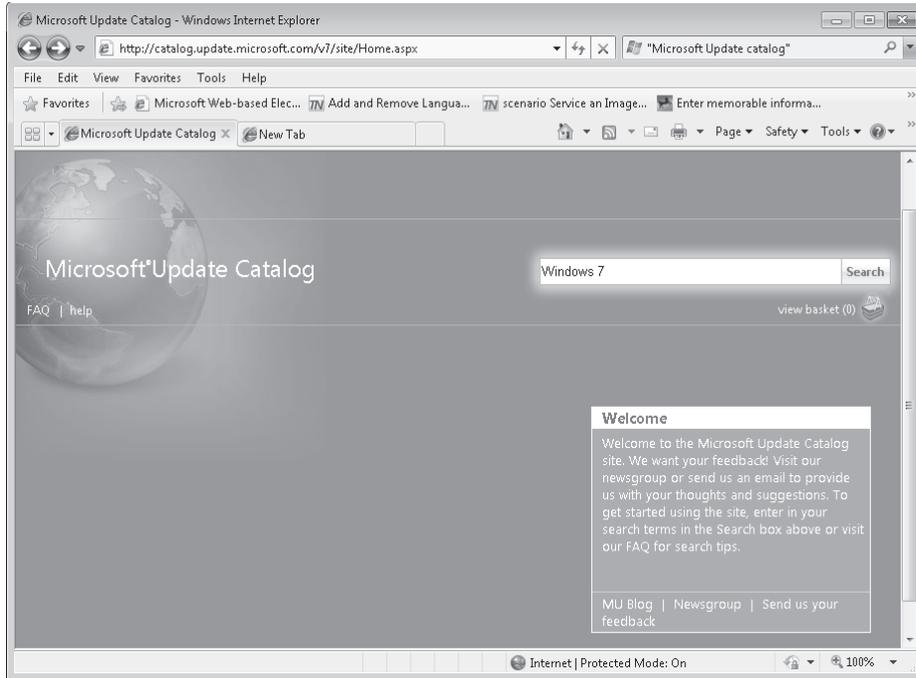


FIGURE 3-30 The Microsoft Update Catalog home page

- **Add updates using WSUS or SCCM 2007** You can use WSUS or SCCM 2007 to install the security updates after deployment. Depending on the configuration, it might take an hour or more before all updates are applied. Including the SCCM client in the image and setting it to communicate with a specific SCCM site can result in all computers built from the image communicating with only that site.
- **Slipstream updates to the installation source** You can download security updates from the Microsoft Update Catalog and integrate them into the Windows installation source before beginning the unattended build process. This protects the image from known security exploits, but integrating the security updates requires administrative effort.

## Keeping an Offline File on a VHD Up to Date

You can use the Offline Virtual Machine Servicing Tool, discussed in Chapter 2, to keep offline VHD files that contain installations of Windows 7 up to date with service packs and software updates. The Offline Virtual Machine Servicing Tool can update a large number of offline virtual machines or VHDs according to their individual needs. The tool works with SCVMM 2007 or SCVMM 2008, in addition to WSUS 3.0, SCCM 2007, or Configuration Manager 2007 R2.

The tool uses the concept of “servicing jobs” to manage the update operations based on lists of existing virtual machines stored in SCVMM. A servicing job runs Windows PowerShell scripts to work with virtual machines and VHDs. The servicing job deploys a virtual machine

to a host and starts it or boots a computer that holds an image installed to implement failover from that image, triggers the software update cycle, and closes down the updated device. The Offline Virtual Machine Servicing Tool then either shuts down the virtual machine or boots the computer that has the VHD installed from its normal boot image.

To use the tool, you configure virtual machine (or VHD) groups and create and schedule servicing jobs. You can schedule jobs to run immediately, or to run during low-traffic maintenance windows. You can also schedule servicing jobs to recur at regular intervals. The disadvantage of the Offline Virtual Machine Servicing Tool is that a virtual machine or physical machine with a bootable VHD is brought online in an insecure state, if only for a short time while the image is updated.

#### **MORE INFO OFFLINE VIRTUAL MACHINE SERVICING TOOL AND SCVMM**

For more information about the Offline Virtual Machine Servicing Tool, see <http://technet.microsoft.com/en-us/library/cc501231.aspx>. For more information about SCVMM 2008, go to <http://technet.microsoft.com/en-us/library/cc668737.aspx> and access the links on the navigation pane.

## **Adding Language Packs**

Language packs create a multilingual Windows environment. Windows operating systems are language-neutral, and language and locale resources are added through language packs (lp.cab files). By adding one or more language packs to Windows 7, these languages can be activated when installing the operating system. As a result, the same Windows 7 image can be deployed to regions with different language and locale settings, reducing development and deployment time.

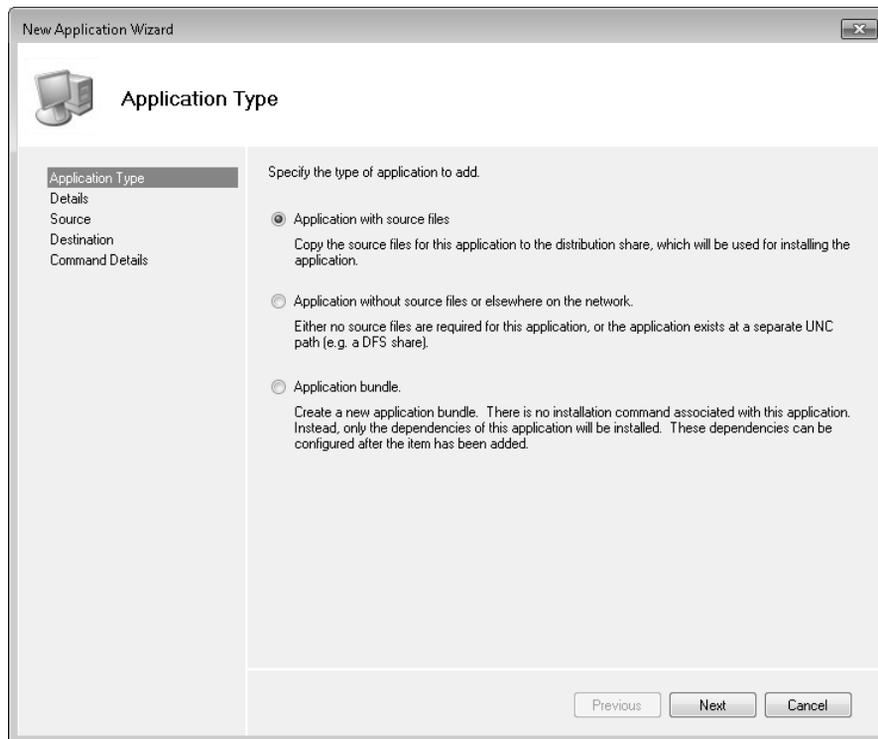
You can add language packs offline or online using MDT 2010 and SCCM 2007. In the Deployment Workbench Task Sequence Editor select the Install Language Packs Offline or Install Language Packs Online task. You are presented with a list of language packs to add. If SCCM 2007 is not available, you can add language packs with a custom task sequence by choosing a template that contains the Add Packages step.

## **Adding Applications**

If you are using a reference computer, you can install applications on that computer and then create an image. Take care that you do not violate licensing conditions if you then install the image on other computers.

You can also add applications to an existing image build by adding them to the distribution share. Deployment Workbench can install the application from its original network location, or it can copy the application source files to the distribution share. In either case, you can specify the commands for installing the application when adding it to the distribution share. Applications can also be installed as SCCM 2007 packages for ZTI deployments. After you have added an application to the distribution share, it can be installed in one of the following ways:

- **Add it to the task sequence** Application installations added to the task sequence occur when MDT 2010 executes the task sequence on the target computer. Typically, for a third-party OEM application, you would choose the LiteTouch OEM Task Sequence template and specify the Copy CD to Local Hard Disk For OEM Pre-Installation step.
- **Use The New Application Wizard** You access this wizard by expanding Distribution Share, right-clicking Applications, and clicking New in the Actions pane. Figure 3-31 shows the Application Type page of the New Application Wizard. In this wizard, you specify the application name and publisher, the source directory for the application files, whether you want to move or copy these files, the name of the destination directory, and the command-line command used to install the application.



**FIGURE 3-31** The New Application Wizard

**CAUTION DO NOT ALLOW AN APPLICATION TO RESTART THE COMPUTER**

If you are using MDT 2010, do not allow an application to restart the computer. MDT 2010 must control restarts, or the task sequence will fail. You can use the command-line property `reboot=reallysuppress` to prevent applications from restarting.



### EXAM TIP

You cannot add an application to an image using DISM. You can, however, add an application to an image build in a distribution share in MDT 2010.

## Configuring Deployment Points

A distribution share contains the files necessary to install and configure a build on a target computer. A deployment point defines a subset of those files and how to connect to them. For example, the distribution share might contain several operating systems and applications. A deployment point defines which of those files to distribute and how to access them.

To create a deployment point, you click Deployment Points in Deployment Workbench and then click New in the Actions pane. The Choose Type page of the New Deployment Point Wizard, shown in Figure 3-32, lets you choose one of the following deployment point types:

- **Lab or single-server deployment point** This enables you to use the distribution share to deploy task sequences.
- **Separate Deployment share** This creates a new local or remote deployment share that contains a subset of the files in the distribution share. You can choose the images, device drivers, updates, and applications that are replicated to this type of deployment point.
- **Removable media** This creates directories and (optionally) an International Organization for Standardization (ISO) image that can be installed on removable media such as DVD-ROM, universal serial bus (USB) disk, or USB flash memory so you can perform stand-alone, network-disconnected deployments.

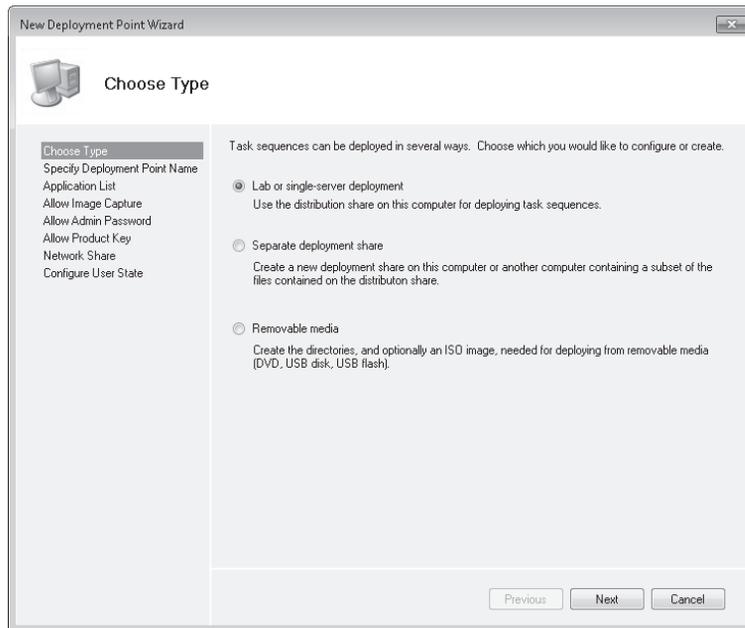


FIGURE 3-32 Choosing the deployment point type

WIM image files and ISO Windows PE image files are created for each deployment point. Client computers connect to the deployment point and the installation begins. During the deployment process, you can choose which build to install from the deployment point.

After you have chosen the type of deployment point, you can specify the deployment point name. Next, you can specify whether to allow users to select additional applications. This control applies in an upgrade scenario where users are typically prompted to install additional applications, but you may want to prevent this because of compatibility considerations.

Typically, if you are deploying a new computer (bare metal deployment) into a workgroup, the deployment wizard asks if an image should be captured. If this is not required, you can configure the deployment point to block this prompt. You can also specify whether users should be prompted for a local administrator password. In a typical scenario, it is considered insecure to permit users to know local administrator passwords. You can also decide whether to prompt users for an installation or activation product key.

The wizard then prompts you for a network share. You need to supply the name of the computer that hosts the distribution share, the share name, and the share path. Finally, you are prompted to configure the user state, which is the location in which information about the user and user settings are stored. By default, this location is determined automatically. Figure 3-33 shows the available options.

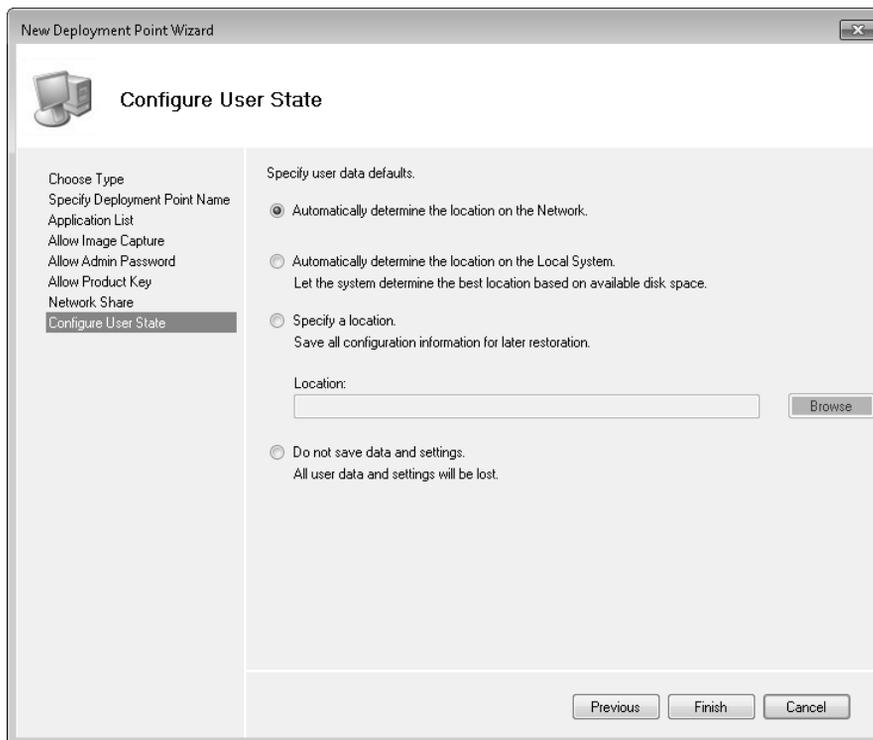


FIGURE 3-33 Specifying the user state

When you have completed the configuration, click Finish to create the deployment point.

**NOTE CONFIGURING A DEPLOYMENT DATABASE**

You can use the New DB Wizard in Deployment Workbench to configure a deployment database. To do this, you need a server running SQL Server 2005 or SQL Server 2008 on your network. This functionality is used when MDT 2010 works with SCCM 2007.

## Configuring Windows PE Options

After creating your deployment point, you need to configure its Windows PE configuration options. Assuming you have configured a LAB deployment point, you do this in Deployment Workbench as follows:

1. In the Deployment Workbench console tree, expand Deploy and select Deployment Point.
2. In the details pane, click LAB.
3. In the actions pane, click Properties.
4. In the LAB Properties dialog box, on the Windows PE tab, in the Driver group, select the device driver group you created earlier in the deployment process (for example, Windows 7) and then click OK.
5. In the details pane, right-click LAB and choose Update.

This updates the deployment point and creates a Windows PE directory. All the MDT 2010 configuration files are updated, and Deployment Workbench generates a customized version of Windows PE that is used to initiate the LTI deployment process.

Deployment Workbench creates the LiteTouchPE\_x86.iso and LiteTouchPE\_x86.wim files (for 32-bit target computers) in the C:\Distribution\Boot folder (where C:\Distribution is the shared folder used as the deployment point share).

## Creating LTI Bootable Media

To boot a reference computer and create an image for distribution, you need to create bootable media containing the customized version of Windows PE that you created when the deployment point was updated. You can create the appropriate LTI bootable media from the LiteTouchPE\_x86.iso or the LiteTouchPE\_x86.wim file. If the reference computer is a physical computer, you can create a bootable DVD ROM from the ISO file. If it is a physical computer with a bootable VHD, you can copy the WIM file in to the VHD. If it is a virtual machine, you can start it directly from the ISO file.

The reference computer boots from the LTI bootable media into Windows PE and the Windows Deployment Wizard starts. You follow the steps of this wizard, specifying details such as your logon credentials, whether the computer is part of a workgroup or domain, and so on. When the wizard completes, a Windows 7 operating system, complete with any additions and amendments you made to the original installation image, is installed on the reference machine.

You need to test the reference computer thoroughly. When you are satisfied that the installation is satisfactory, you can create an image as described in Chapter 2 and deploy it with either MDT or WDS.

If your target computers are not PXE-compliant, you boot them from the LTI bootable media. Microsoft recommends that you do not do this for PXE client computers but instead use WDS with MDT 2010 to deploy these computers through LTI. WDS is listed as required software to enable MDT 2010 to implement LTI, but only if you are deploying PXE-compliant computers.

## Deploying Images with WDS

Chapter 2 discussed WDS and WDS images. WDS is installed as a server role and deploys images to multiple computers. An advantage of using WDS is that it uses multicast transmissions. As a result, an operating system image needs to be transferred across the network only once to be deployed to multiple computers.



### **EXAM TIP**

Although WDS is a server role, the topic is prominent in the 70-680 examination objectives, and it is likely to be tested.

## Installing and Configuring WDS

You install WDS as a server role on a server running Windows Server 2008 or Windows Server 2008 R2 that is a member of an Active Directory Domain Services (AD DS) domain. Because WDS deploys to clients that are PXE-compliant, you must have a Dynamic Host Configuration Protocol (DHCP) server on your network. You also require a Domain Name System (DNS) server and your WDS deployment server requires an NTFS file system volume for its image store. You must be a member of the Local Administrators group on the server. To use WDS to deploy images, you need to select the Deployment Server option when installing the server role.

After you install the server role, you must configure the server, add a boot image, and add an install image. The server will then be ready to deploy images to target computers.

The high-level procedure to configure the WDS server role is as follows:

1. Open the Windows Deployment Services console from the Administrative Tools menu. If there is no server listed in the Servers node, right-click the node and choose Add Server to add the local server.
2. In the left pane of the Windows Deployment Services console, expand the server list.
3. Right-click the local server, and then choose Configure Server.
4. Follow the instructions in the wizard.
5. When the configuration completes, clear the Add Images To Windows Deployment Services Now check box and then click Finish.
6. If you want to modify any of the settings of the server, right-click the server in the console, and choose Properties.

## Adding Boot and Install Images

After you have configured the server, you need to add images. These images include a *boot image* (the bootable environment that you initially boot a target computer into), and one or more *install images* (the images that you deploy). Initially you add the default boot image (Boot.wim) included on the Windows Server or Windows 7 installation DVD-ROM. The Boot.wim file contains Windows PE and the WDS client. The high-level procedure to add the default boot image is as follows:

1. In the left pane of the Windows Deployment Services console, right-click the Boot Images node, and then choose Add Boot Image.
2. Select the default boot image (Boot.wim) in the \Sources folder on the Windows Server installation DVD-ROM.
3. Click Open and then click Next.
4. Follow the instructions in the wizard to add the image.

Install images are the operating system images that you deploy to the client computer. For Windows 7, you can also use the Install.wim file from the Windows 7 installation DVD, or you can create your own install image from a reference computer running Windows 7. WDS can use a capture image to capture the image of a reference computer. The high-level procedure to add the default install image from a Windows 7 installation DVD-ROM (Install.wim) is as follows:

1. In the Windows Deployment Services console, right-click the Install Images node and choose Add Install Image.
2. Specify an image group name and click Next.
3. Select the default install image (Install.wim) in the \Sources folder on the Windows 7 DVD-ROM and click Open.
4. If you do not want to add all the images in Install.wim on the DVD-ROM, clear the check boxes for the images that you do not want to add. Add only the images for which you have licenses.
5. Follow the instructions in the wizard.

## Deploying an Install Image

You can now deploy the install image directly to PXE-compliant target computers. In practice, you would not install the image from the DVD-ROM directly to a number of target computers, which would make these computers vulnerable to known security threats.

You could update the image with security patches, drivers, language packs, and so on with a tool such as DISM, or you could use WDS with MDT 2010, which can add security patches, language packs, and applications. Even then, you would deploy to only one reference computer and test it carefully before deploying it across the enterprise. If you make any changes to your reference computer, you can use a capture image to capture the amended settings on the reference computer.

The high-level procedure to deploy an install image to a PXE-compliant target computer is as follows:

1. Configure the BIOS of the target computer to enable PXE booting, and set the boot order so that it is booting from the network first.
2. Restart the computer, and when prompted, press F12 to start the network boot.
3. If you have more than one boot image on the WDS server, you are presented with a boot menu on the client. Select the appropriate boot image.
4. Follow the instructions in the Windows Deployment Services user interface.

When the installation is complete, the target computer restarts and Setup continues.

## Creating a Discover Image

If you need to deploy a Windows 7 operating system to a computer that is not PXE-compliant, you should create a discover image and save it to bootable media such as a DVD-ROM or bootable USB flash drive. Booting the target computer from the discover image enables it to locate a WDS server, which then deploys the install image to the computer. You can configure discover images to target a specific WDS server. If you have multiple WDS servers in your environment, you can create a discover image for each one.

You can create a discover image from the Boot.wim file on the Windows Server 2008 or Windows 7 installation DVD-ROM. You cannot use the Windows PE file (WinPE.wim) from Windows AIK to create a discover image. Note, however, that Windows AIK needs to be installed on the WDS server to create the bootable media that contains the discover image. The high-level procedure to create a discover image and install it on bootable media is as follows:

1. In the Windows Deployment Services console, expand the Boot images node.
2. Right-click the image that you want to use as a discover image. This must be the Boot.wim file from the Windows Server or Windows 7 DVD-ROM.
3. Click Create Discover Boot Image.
4. Follow the instructions in the wizard, and when it is completed, click Finish.
5. To create media that contains the discover image, click Microsoft Windows AIK in the All Programs menu and then download and install the Windows AIK (<http://www.microsoft.com/downloads/details.aspx?FamilyId=94BB6E34-D890-4932-81A5-5B50C657DE08&displaylang=en>).
6. Click Start, click All Programs, and then click Windows PE Tools Command Prompt.
7. To create a Windows PE build environment, enter the following:

```
copype architecture c:\winpe
```

8. To copy the discover image that you created, enter the following:

```
copy /y c:\imagename.wim c:\winpe\iso\sources
```

9. To change back to the PETools folder, enter the following:

```
cd c:\program files\windows aik\tools\petools
```

10. To create the bootable .iso image, enter the following:

```
oscdimg -n -bc:\winpe\iso\boot\etfsboot.com c:\winpe\iso c:\imagename.iso
```

11. Create a bootable DVD-ROM or USB flash drive from the ISO image. If you transfer the image to a Windows 7 (or Windows Vista) client, double-clicking the image does this for you. Otherwise, use reputable third-party software.

## Creating a Capture Image

Capture images are boot images into which you boot a client computer to capture its operating system in a WIM file. You create a capture image, run Sysprep on the reference computer, restart the reference computer, press F12 (or use a discover image if the reference computer is not PXE-compliant), select the capture image which should now appear on the boot menu, capture the reference computer image as a WIM image, and upload it to the WDS server.

Note that you can capture a system image using the ImageX tool in the Windows AIK and install it on the WDS server, but a capture image automates the process. Typically, you create a capture image from Boot.wim. The high-level procedure to do this is as follows:

1. In the Windows Deployment Services console, expand the Boot Images node.
2. Right-click the image you want to use as a capture image (typically, Boot.wim).
3. Choose Create Capture Boot Image.
4. Type a name, a description, and the location where you want to save a local copy of the file. You specify this location in case there is a network problem when you deploy the capture image.
5. Follow the instructions in the wizard, and when it is complete, click Finish.
6. Right-click the boot image folder.
7. Choose Add Boot Image.
8. Select the new capture image, and then click Next.
9. Follow the instructions in the wizard.

## WDS Images

In the previous sections, we looked at how WDS creates install, boot, capture, and discover images. However, it is valuable at this juncture to briefly summarize the purpose of these images. WDS installs an install image (typically a WIM file) to its target computers. It cannot manipulate this file by adding drivers, language packs, and applications (for example) to its distribution share as can MDT 2010, but you can manipulate the WIM image with DISM before you distribute it with WDS. You can also deploy the WDS image to a reference computer, test and amend it online if necessary, ensure it is up to date, generalize it using Sysprep, and then use a capture image to create an install image on the WDS server.

WDS works by first booting the target computers with a boot image. This enables the deployment of the install image to the target computers.

WDS can also capture an image on a reference computer and install it to multiple target computers. To do this, the reference computer boots from a special type of boot image called a capture image. A capture image contains Windows PE and the Windows Deployment Services Image Capture Wizard. When the reference computer boots from a capture image (after you prepare it with Sysprep), the wizard creates an install image of the computer and saves it as a WIM file. This then becomes the install image that WDS deploys to the target computers after booting them with a standard boot image.

Typically WDS works with PXE-enabled computers. PXE enables computers to boot from the network to a state that allows you to select a WDS boot image. If the target computers are not PXE-enabled, WDS can deploy an operating system provided the computers are first booted with a discover image. When you boot a computer into a discover image, the WDS client locates a valid WDS server, and you can choose the image you want to deploy.

**NOTE WDS AND STANDARD WINDOWS PE IMAGES**

The WDS discover image contains a Windows PE image and WDS client software. You should not boot into Windows PE from a Windows PE boot disk (for example) to attempt to access a WDS server. You should not attempt to boot to Windows PE or from a discover image if your target computer is PXE-compliant.

## Booting a Target Computer Manually

If a computer is not PXE-compliant and you need to boot it manually using a discover image, you can use optical media or other removable media, such as a USB hard disk or USB flash memory. You can use the BCDEdit and BCDBoot tools to create bootable media. Chapter 2 discussed the BCDEdit tool and used it to mark a VHD as bootable. You can use the same procedure for USB devices.

BCDboot is a Windows AIK tool that you can use to set up a system partition or to repair the boot environment located on the system partition. The tool creates a system partition by copying a small set of boot environment files from an installed Windows image. BCDboot also creates a BCD store on the system partition with a new boot entry that enables you to boot to the installed Windows image. You can run BCDboot from Windows PE. If you have not installed the Windows AIK, you still have access to BCDboot in the Windows\System32 directory. The tool is also available in the Windows OEM Preinstallation Kit (OPK).

Specific BCD settings can be defined in the BCD-Template file. The tool also copies the most recent versions of boot-environment files from the operating system image Windows\Boot folder and Windows\System32 folder to the default system partition identified by the firmware. You can create this partition by using a partitioning tool such as DiskPart.

When a computer has been booted into Windows PE and an image is installed, you use BCDBoot to initialize the BCDstore and copy boot environment files to the system partition. This allows the computer to boot normally when it is restarted without Windows PE.

#### **MORE INFO DISKPART**

For more information about Diskpart, see <http://support.microsoft.com/kb/300415>. Although this article was written for Windows XP, it has been updated since and is relevant to Windows 7. The Diskpart tool is discussed in detail in Chapter 4, “Managing Devices and Disks.”

#### **MORE INFO BCDBOOT**

For more information about BCDboot, see <http://support.microsoft.com/kb/300415>.

## Using the WDSUTIL Tool

WDS provides a GUI tool and a command-line tool. You can perform most deployment tasks using the Windows Deployment Services console in the Microsoft Management Console (MMC), including setting the Auto-Add policy and approving or rejecting pending computers. However, if you want to prestage client computers, you need to use the WDSUTIL command-line tool.

For example, you can use the WDSUTIL */add* construct to add images or image groups, or to prestage computers. The following command prestages the computer Aberdeen with a MAC address 00-13-E8-64-46-01:

```
WDSUTIL /Add-Device /Device:Computer1 /ID: 00-13-E8-64-46-01
```

The following command adds a boot image to the WDS server. The image is called Myboot.wim and is stored on C:\MybootImages.

```
WDSUTIL /Add-Image /ImageFile:"C:\mybootimages\myboot.wim" /ImageType:Boot
```

You can use the WDSUTIL tool to carry out the following tasks:

- Configure a WDS server for initial use.
- Start and stop all services on the WDS server.
- Update serve files on the RemoteInstall share.
- Revert changes made during server initialization.
- Create new capture and discover images, as well as multicast transmissions and namespaces.
- Add images and image groups and prestage computers.
- Approve or reject computers that are pending administrator approval.
- Copy an image within the image store.
- Export an image from the image store to a .wim file.
- Replace a boot or installation image with a new version of that image.
- Remove images, image groups, multicast transmissions, and namespaces.

- Convert an existing Remote Installation Preparation (RIPrep) image to a Windows Image (.wim) file.
- Delete computers that are in the Auto-Add Device Database, which stores information about the computers on the server.
- Disable or enable all services for WDS.
- Disconnect a client from a multicast transmission.
- Set properties and attributes of a specified object.
- Retrieve properties and attributes of a specified object.
- Display the progress status while a command is being executed.

#### **MORE INFO WDSUTIL**

For more information about WDSUTIL, including syntax and code examples, see <http://technet.microsoft.com/en-us/library/cc771206.aspx>.

#### **MORE INFO WDS**

For more information about WDS, go to <http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=3cb929bc-af77-48d2-9b51-48268cd235fe> and download the WDS documentation files.

## Using SCCM 2007

SCCM 2007 (ConfigMgr) is Windows Server 2003 or Windows Server 2008 software that implements change and configuration management for Microsoft platforms. It enables you to perform tasks such as the following:

- Deploying operating systems, software applications, and software updates
- Metering software usage
- Assessing variation from desired configurations
- Taking hardware and software inventory
- Remotely administering computers

SCCM 2007 collects information in a SQL Server database that you can configure, using tools such as MDT 2010. This allows queries and reports to consolidate information throughout the organization. SCCM 2007 can manage a wide range of Microsoft operating systems, including client platforms, server platforms, and mobile devices. It works with MDT 2010 to implement ZTI.

SCCM 2007 collects hardware and software inventories, distributes and installs software applications and software updates such as security fixes. It works with Windows Server 2008 Network Policy Server (NPS) to restrict computers from accessing the network if they do not meet specified requirements, such as having security updates installed. SCCM 2007

determines what a desired configuration should be for one or more computers, and monitors adherence to that configuration. It controls computers remotely to provide troubleshooting support.

### ✓ Quick Check

- What command-line utility enables you to prestage target computers for system image deployment?

### Quick Check Answer

- WDSUTIL

## SCCM Clients and Sites

After you have installed your central SCCM 2007 site, you can add clients and resources to the site. These are added by using one of the available SCCM discovery methods, which search your network to find resources that you can use with SCCM. You must discover computers on your network before you can install the Configuration Manager client software that allows you to deliver such items as packages and updates to those clients. Discovery methods include:

- Heartbeat Discovery
- Network Discovery
- Active Directory User Discovery
- Active Directory System Group Discovery
- Active Directory Security Group Discovery
- Active Directory System Discovery

### **MORE INFO** SCCM CLIENT DISCOVERY

In-depth discussion of client discovery is beyond the scope of the 70-680 examination and this book. If you want to learn more out of professional interest, see <http://msdn.microsoft.com/en-us/library/cc143989.aspx>.

When it has discovered clients, SCCM 2007 installs client software on the Windows-based computers it manages. Configuration Manager 2007 client software can be installed on desktop computers, servers, portable computers such as laptops, mobile devices running Windows Mobile or Windows CE, and devices running Windows XP Embedded (for example, automated teller machines).

You can use SCCM 2007 to group clients into sites. SCCM sites group clients into manageable units with similar requirements for feature sets, bandwidth, connectivity, language, and security. SCCM 2007 sites can match your AD DS sites or be totally independent of them. Clients can move between sites or be managed from remote locations such as home offices.

Clients communicate with site systems hosting site system roles. Site systems communicate with the site server and with the site database. If there are multiple sites connected in a hierarchy, the sites communicate with their parent, child, or, sometimes, grandchild sites. SCCM 2007 uses boundaries to determine when clients and site systems are in the site and outside the site. Boundaries can be IP subnets, IP address ranges, IPv6 prefixes, and AD DS sites. Two sites should never share the same boundaries.

When SCCM 2007 features within the same site communicate with each other, they use either server message block (SMB), Hypertext Transfer Protocol (HTTP), or Hypertext Protocol Secure (HTTPS), depending on various site configuration choices you make. Because all these communications are unmanaged, it is a good idea to make sure these site elements have fast communication channels.

## SCCM 2007 Task Sequence Editor

SCCM 2007 uses task sequences in a similar way to MDT 2010, and you can export and import task sequences between the tools. The SCCM 2007 Task Sequence Editor creates and modifies task sequences that are organized into groups of task sequence steps. Depending on whether the Install An Existing Image package or the Build A Reference Operating System Image package is selected in the New Task Sequence Wizard, the task sequence contains a set of baseline task sequence groups and steps. If the Create A New Custom Task Sequence is selected in the New Task Sequence Wizard, an empty task sequence is created.

The Task Sequence Editor displays the task sequence groups and steps in the tree view on the left side of the editor window in a manner similar to the MDT 2010 Task Sequence Editor. When you select a task sequence group or step, its properties are displayed next to the tree view with tabs that you can select to configure settings. Task sequence groups and steps can be nested within other task sequence groups.

Task sequence steps are grouped into general, disk, user state, images, drivers, and settings. General SCCM task sequence steps include the following:

- **Run Command Line** This task sequence step can run any command line. The task sequence action can be run in a standard operating system or Windows PE.
- **Install Software** This task sequence step specifies an SCCM 2007 package and program to install as part of the task sequence. The installation will begin immediately without waiting for a policy polling interval. The Install Software task sequence step runs only in a standard operating system such as Windows 7 and will not run in Windows PE.
- **Install Software Updates** This task sequence step installs software updates on a target computer. When this task sequence step runs, the target computer is evaluated for applicable software updates. In particular, the step installs only software updates that are targeted to collections of which the computer is currently a member. The Install Software Updates task sequence step runs only in a standard operating system such as Windows 7 and will not run in Windows PE.

- **Join Domain or Workgroup** This task sequence action adds a target computer to a workgroup or domain. The Join Domain or Workgroup task sequence step runs only in a standard operating system such as Windows 7 and does not run in Windows PE.
- **Connect to Network Folder** This task sequence action creates a connection to a shared network folder. The Connect to Network Folder task sequence step runs only in a standard operating system such as Windows 7 and does not run in Windows PE.
- **Restart Computer** This task sequence step restarts the computer running the task sequence. After the restart, the computer automatically continues with the next step in the task sequence. The Restart Computer task sequence action can be run in either a standard operating system or Windows PE.
- **Set Task Sequence Variable** This task sequence step sets the value of a task sequence variable to be used with the task sequence. Task sequence actions read task sequence variables, which specify the behavior of those actions.

#### **MORE INFO TASK SEQUENCE ACTIONS AND VARIABLES**

For more information about task sequence actions, see <http://technet.microsoft.com/en-us/library/bb632625.aspx>. For more information about task sequence variables, see <http://technet.microsoft.com/en-us/library/bb632442.aspx>.

Disk steps, user state steps, image steps, driver steps, and setting steps let you configure the following on SCCM clients:

- Disk steps
  - Format and partition disk
  - Convert disk to dynamic
  - Enable and disable BitLocker
- User state steps
  - Request state store
  - Release state store
  - Capture user state
  - Restore user state
- Image steps
  - Apply operating system
  - Apply data image
  - Install deployment tools
  - Prepare ConfigMgr client for capture
  - Prepare Windows for capture
  - Capture operating system image

- Driver steps
  - Auto-apply drivers
  - Apply driver package
  - Setting steps
  - Capture network settings
  - Capture Windows settings
  - Apply network settings
  - Apply Windows settings

## Integrating SCCM 2007 and MDT 2010

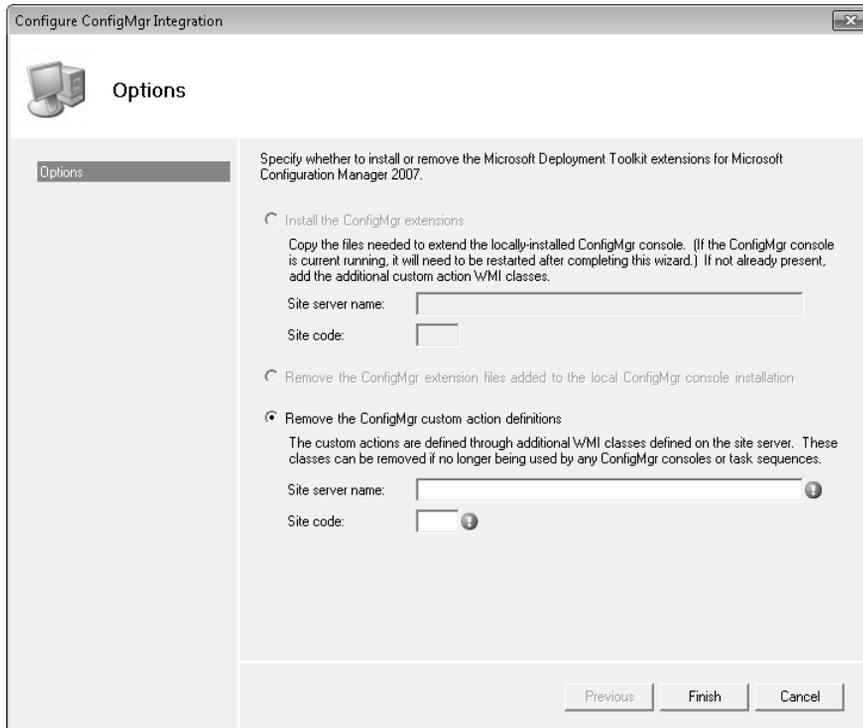
SCCM 2007 and MDT 2010 can be integrated in the Configuration Manager console on a distribution server to implement tasks such as installing language packs. However, possibly the most typical reason for integrating the two tools is that it enables you to implement ZTI. You need to install MDT 2010 on each computer running the Configuration Manager console. The SCCM 2007 Integration option can then be implemented, and data can be specified for MDT 2010 packages.

Before you can use the SCCM integration features of MDT 2010, you need to run the Configure Configuration Manager 2007 Integration script. This script copies the appropriate SCCM integration files to *Configuration Manager 2007\_root* (where *Configuration Manager 2007\_root* is the folder in which SCCM is installed). The script also adds Windows Management Instrumentation (WMI) classes for the new MDT 2010 custom actions. The classes are added by compiling a new Managed Object Format (.mof) file that contains the new class definitions. The .mof file is the mechanism by which information about WMI classes is entered into the WMI Repository.

Before you run the Configure Configuration Manager 2007 Integration script, ensure the Configuration Manager console is closed. The high-level procedure to run this script is as follows:

1. On the All Programs menu, click Microsoft Deployment Toolkit and then choose Configure ConfigMgr 2007 Integration. Figure 3-34 shows the Configure ConfigMgr Options page (as yet unconfigured).
2. In the Site Server Name dialog box, type the name of the SCCM 2007 server on which you want to implement MDT 2010 integration and then click OK.
3. In the Site Code dialog box, type the SCCM site code that installs MDT 2010 integration and then click Finish.

You can now manage your deployment using the features and utilities provided by MDT 2010 with SCCM features such as client discovery and client integration methods to provide a fully automated deployment that requires no user intervention.



**FIGURE 3-34** The Configure ConfigMgr Options page

**MORE INFO SCCM 2007 AND SOFTWARE UPDATE INSTALLATION**

For more information about using SCCM 2007 and the built-in ConfigMgr Install Software Updates task sequence, see <http://technet.microsoft.com/en-us/library/bb632402.aspx>.

## Installing an Image Manually

Sometimes you do not want to use sophisticated deployment tools such as MDT 2010, SCCM 2007, or WDS. Suppose, for example, you have installed a computer running Windows 7 on your small office/home office (SOHO) network, generalized the installation using Sysprep, and created a bootable Windows PE DVD-ROM disk (or bootable USB hard disk or flash memory) by using the Copype.cmd script.

You have copied the ImageX tool into the Iso subdirectory on the Windows PE media, booted your computer into Windows PE, and used ImageX to create a WIM image of your computer installation. You have booted into the Windows PE environment and used ImageX to capture an image of your computer. You have copied the resulting WIM file on to your Windows PE media (and you might have used DISM to add additional drivers if you wanted).

You now want to apply this customized image to the hard disks of two new computers you have purchased without operating systems. You boot each computer in turn from the Windows

PE media and use ImageX to install the image. Your final step, to make the image bootable, is to use BCDboot from Windows PE to initialize the BCD store and copy boot environment files to the system partition. When you reboot each new computer, it will boot into Windows 7 and will have the same settings configured and applications installed as your original computer. Take care you are not violating any licensing conditions.

## **PRACTICE** Downloading, Installing, and Configuring MDT 2010

In this practice, you download the MDT 2010 installation and documentation files and then install the toolkit. You use the Deployment Workbench tool to create a Distribution Share and install an image.

### **NOTE THE MDT 2010 INTERFACE**

At the time of this writing, MDT 2010 is in beta. Therefore, its eventual interface might vary from what you see in this book.

### **EXERCISE 1** Downloading the MDT 2010 Installation Files and Documentation

In this exercise, you download the MDT and its associated documentation by accessing <https://connect.microsoft.com/site/sitehome.aspx?SiteID=14>. You probably first need to supply your Microsoft password credentials. You have the option of downloading the following files:

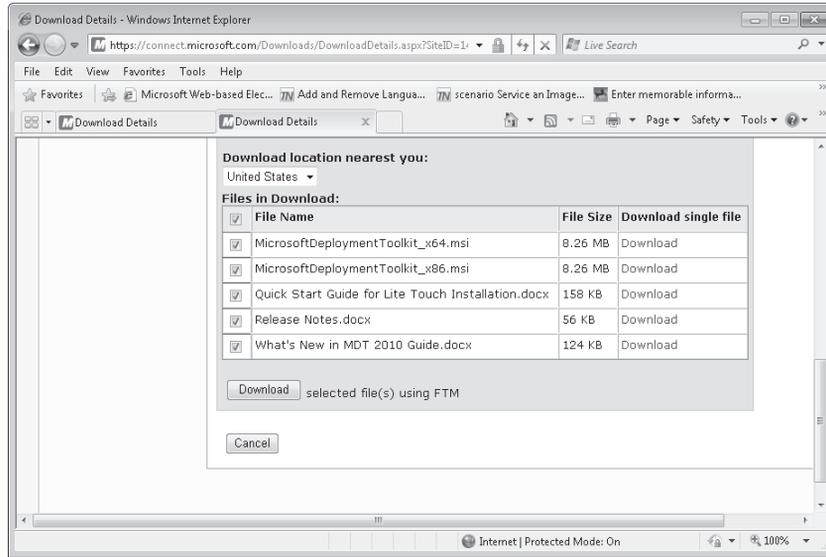
- MicrosoftDeploymentToolkit\_x64.msi
- MicrosoftDeploymentToolkit\_x86.msi
- Quick Start Guide for Lite Touch Installation.docx
- Release Notes.docx
- What's New in MDT 2010 Guide.docx

You can download and install the version suitable for your operating system—this book assumes the 32-bit (x86) version. You need no additional software to run the MDT on Windows 7, although if you choose to use the MDT in conjunction with SCCM 2007 on a deployment server, you need to install the relevant software and additional software such as SQL Server.

To download MDT 2010 and its associated documentation, proceed as follows:

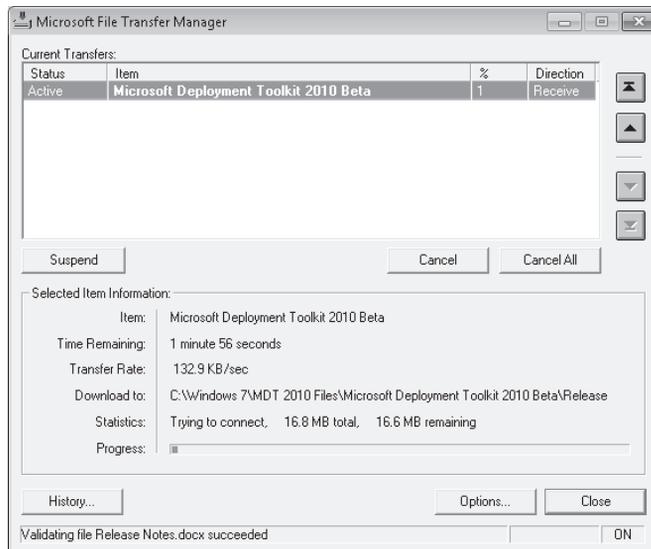
1. Log on to the Canberra computer with the Kim\_Akers account.
2. Create a folder to hold the downloaded files; for example, C:\Windows 7\MDT 2010 Files. Also create a folder to hold documentation, such as C:\Windows 7\MDT 2010 Documentation.
3. Open Internet Explorer and access <https://connect.microsoft.com/site/sitehome.aspx?SiteID=14>. If asked, supply your Microsoft Password details.
4. Click Microsoft Deployment Toolkit 2010.
5. Under Microsoft Deployment Toolkit (MDT) 2010, click Download.

- Specify the files that you want to download and the Download Location Nearest You, as shown in Figure 3-35. Click Download.



**FIGURE 3-35** Selecting files to download

- If necessary, click Allow to open Web content. Also, if prompted, right-click the Address Bar and install the required ActiveX control.
- Ensure that the files received will be placed in the folder C:\Windows 7\MDT 2010 Files. If not, browse to that folder.
- Click Transfer. Microsoft File Transfer Manager transfers the files. Figure 3-36 shows the transfer. Click Close when the transfer completes.



**FIGURE 3-36** Microsoft File Transfer Manager transfers the files.

10. Access C:\Windows 7\MDT 2010 Files. Check that all the specified files have been downloaded to the subfolder folder Microsoft Deployment Toolkit 2010 Beta.

### EXERCISE 2 Accessing the MDT 2010 Documentation

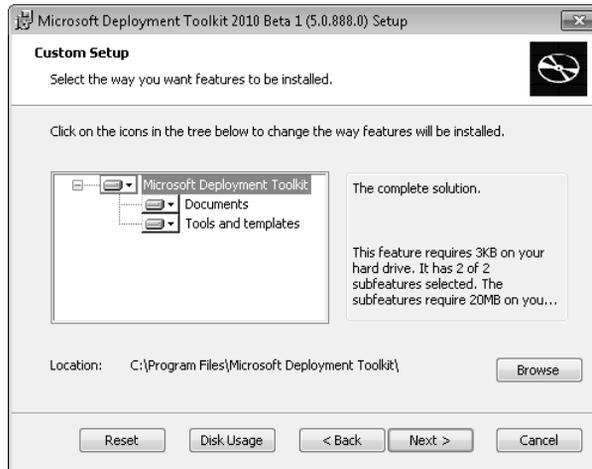
Microsoft recommends that you read the MDT 2010 documentation before installing the tool. To access this documentation, proceed as follows:

1. If necessary, log on to the Canberra computer with the Kim\_Akers account.
2. Access the folder C:\Windows 7\MDT 2010 Files. Open the Microsoft Deployment Toolkit 2010 Beta folder.
3. Double-click Quick Start Guide For Lite Touch Installation.docx and save the file to C:\Windows 7\MDT 2010 Documentation.
4. Double-click Release Notes.docx and save the file to the same location.
5. Double-click What's New In MDT 2010 Guide .docx and save the file to the same location.
6. Read the downloaded documents carefully before you attempt Exercise 3 in this practice session.

### EXERCISE 3 Installing MDT 2010

In this exercise, you install MDT 2010. This exercise assumes that you have installed the x86 version of Windows 7 Ultimate. If you have installed a 64-bit operating system, install the x64 version of MDT 2010 instead. To install MDT 2010, proceed as follows:

1. If necessary, log on to the Canberra computer with the Kim\_Akers account.
2. Right-click the file MicrosoftDeploymentToolkit\_x86.msi, which you downloaded in Exercise 1 and choose Install.
3. On the Welcome page, click Next.
4. On the End-User License Agreement page, review the license agreement. If you agree with the terms, select I Accept The Terms In The License Agreement, and then click Next.
5. On the Custom Setup page, shown in Figure 3-37, choose the features to install and the destination folder for the installation, and then click Next. The features are as follows:
  - **Documents** This feature installs the guidance and job aids. By default, this feature is installed in C:\Program Files\Microsoft Deployment Toolkit\Documentation. The prerequisites for installing this feature and using Deployment Workbench to view the documentation are Microsoft .NET Framework version 2.0 and MMC version 3.0, both of which are implemented in Windows 7.
  - **Tools and templates** This feature installs the wizards and template deployment files, such as Unattend.xml. By default, this feature is installed in C:\Program Files\Microsoft Deployment Toolkit.



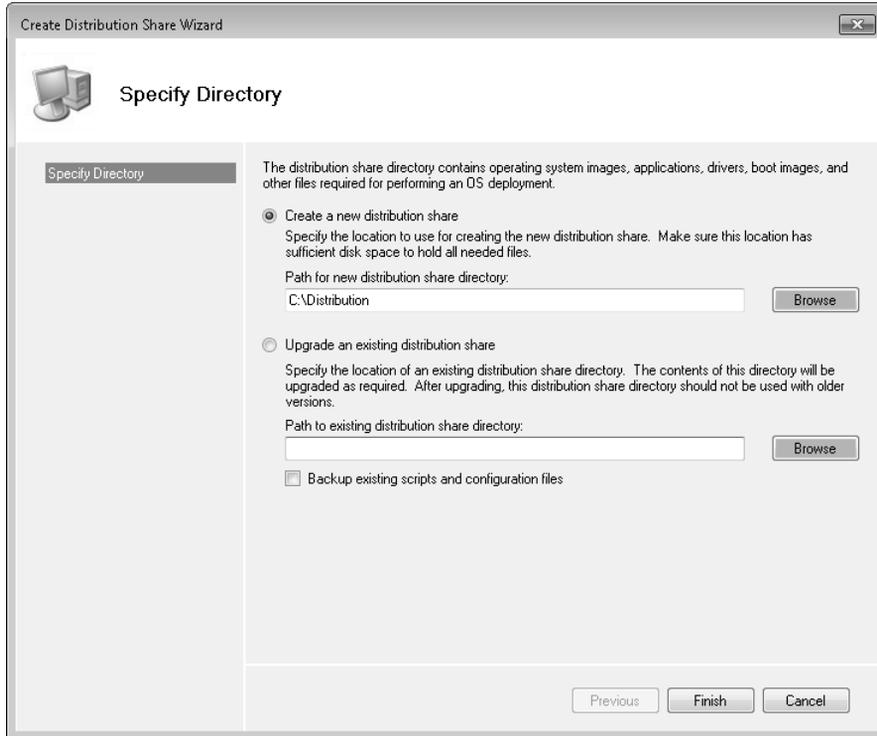
**FIGURE 3-37** Choosing the MDT features to install

6. If you want to change a feature's state, click the feature, and then choose a state. To change the destination folder, click Microsoft Deployment Toolkit, and then click Browse. In the Change Current Destination Folder dialog box, specify the new destination folder and then click OK.
7. Ensure Microsoft Development Toolkit is selected, as shown previously in Figure 3-37, and click Install. If prompted, click Yes to allow the program to continue.
8. When installation completes, click Finish.

#### **EXERCISE 4** Creating a Distribution Share

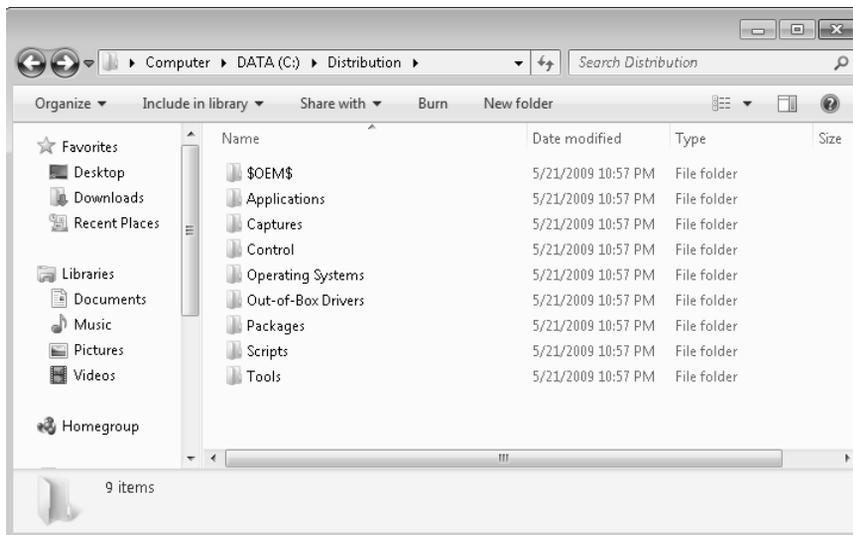
In this exercise, you create a distribution share. You must have installed MDT 2010 before you attempt this exercise. To create a distribution share, proceed as follows:

1. If necessary, log on to the Canberra computer with the Kim\_Akers account.
2. On the Start menu, click All Programs. Click Microsoft Deployment Toolkit and then choose Deployment Workbench. If necessary, click Yes to allow the program to make changes to your computer.
3. In Deployment Workbench, right-click Distribution Share and then choose Create Distribution Share Directory.
4. On the Specify Directory page of the Create Distribution Share Wizard, click Create A New Distribution Share.
5. Type **C:\Downloads** in the Path For New Distribution Share Directory text box, as shown in Figure 3-38, and then click Finish.



**FIGURE 3-38** Specifying the path for the distribution share

6. Navigate to C:\Downloads and check that the folder is populated by subfolders, as shown in Figure 3-39. Some of these folders (such as Applications) are initially empty; others (such as Tools) are not.



**FIGURE 3-39** Contents of a newly created distribution share

## EXERCISE 5 Install an Image on the Distribution Share

In this exercise, you install the Windows 7 source files on the Windows 7 Installation media in the distribution share that you created in the previous exercise. To install this image, proceed as follows:

1. If necessary, log on to the Canberra computer with the Kim\_Akers account.
2. Insert the Windows 7 x86 installation DVD-ROM. If necessary, close the AutoPlay dialog box. As this exercise is written, the DVD-ROM drive is F:. If your computer uses a different drive letter, amend the procedure accordingly.
3. If necessary, open Deployment Workbench.
4. Under Deployment Workbench, expand Distribution Share and select Operating Systems.
5. In the Action pane, click New.
6. Ensure that Full Set Of Source Files is selected on the OS Type page of the New OS Wizard, as shown in Figure 3-40. You could also specify a Custom Image file, such as the Myimage.wim file that you captured and deployed on your bootable VHD in Chapter 2 or a WDS image available on a specified WDS server.

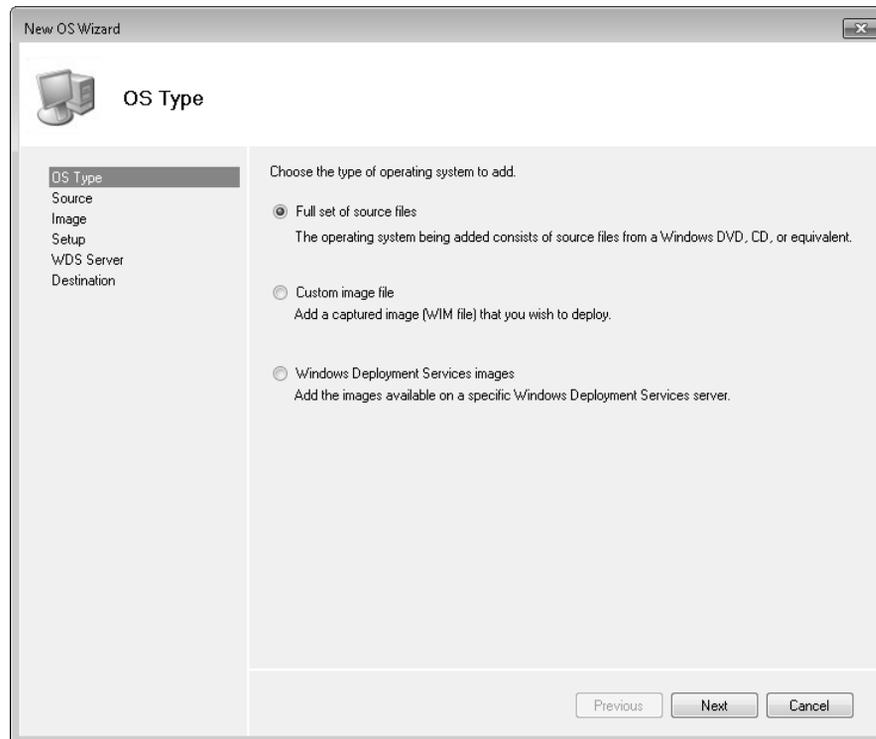
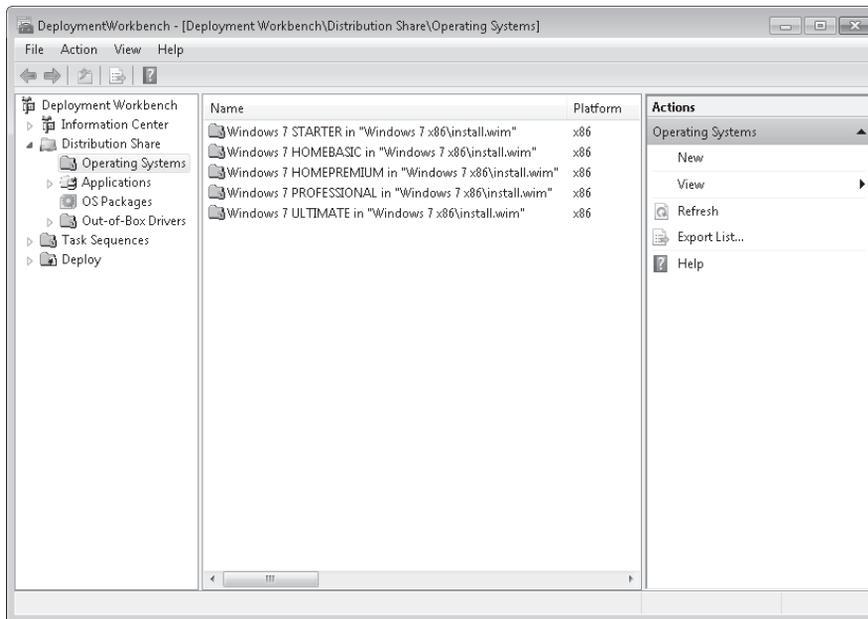


FIGURE 3-40 Specifying a full set of source files

7. Click Next.

8. On the Source page, type **F:** in the Source Directory text box. Click Next.
9. Ensure that Windows 7 x86 is entered in the Destination Directory Name text box and click Finish.
10. The copy operation takes some time. When it is complete, ensure that the appropriate operating system images have been placed in the distribution share, as shown in Figure 3-41.



**FIGURE 3-41** Operating system images placed in the distribution share

## Lesson Summary

- MDT 2010 enables you to manage and manipulate disk images and to create a distribution share to distribute an operating system image to other computers on your network. You need to install Windows AIK before you can create or deploy WIM image files.
- WDS uses boot images that enable PXE-compliant computers to boot from the network and obtain an install image. If a computer is not PXE-compliant, you can boot it from a discover image on bootable media and WDS can then deploy an install image to it. Discover images enable you to boot a reference computer and transfer its system image to WDS, which can then deploy it to other computers.
- MDT 2010 can work with WDS in an LTI scenario. To implement ZTI, MDT 2010 requires that SCCM 2007 and SQL Server are available on the network. MDT 2010 requires that Windows AIK is installed.

## Lesson Review

You can use the following questions to test your knowledge of the information in Lesson 2, “Deploying Images.” The questions are also available on the companion DVD if you prefer to review them in electronic form.

### **NOTE ANSWERS**

Answers to these questions and explanations of why each answer choice is correct or incorrect are located in the “Answers” section at the end of the book.

- 1.** You want to ensure that offline VHD files that contain installations of Windows 7 are kept up to date with service packs and software updates. Which tool should you use?
  - A.** MDT 2010
  - B.** Offline Virtual Machine Servicing Tool
  - C.** BCDEdit
  - D.** Configuration Manager 2007 R2
- 2.** You want to deploy a WIM image file captured from a reference computer running Windows 7. You need to specify the source directory in which the WIM file resides, specify whether Setup or Sysprep files are required, and then move the file to the distribution share. What tool lets you do this?
  - A.** The Windows Deployment Services Image Capture Wizard, which is contained in a WDS capture image
  - B.** The New Task Sequence Wizard, which you can access from the SCCM 2007 Task Sequence Editor
  - C.** The Create Distribution Share Wizard, which you can access from the Deployment Workbench console
  - D.** The New OS Wizard, which you can access from the MDT 2010 Deployment Workbench console
- 3.** WDS creates install, boot, discover, and capture images. Which of these do you need to install on bootable removable media?
  - A.** Install
  - B.** Boot (standard boot image)
  - C.** Discover
  - D.** Capture
- 4.** Which of the following are required if WDS is to be installed and to deploy images? (Choose all that apply.)
  - A.** AD DS
  - B.** MDT 2010

- C.** SQL Server
  - D.** SCCM 2007
  - E.** DHCP
  - F.** DNS
- 5.** You have created a bootable DVD-ROM containing a Windows PE image, the ImageX tool, and a Windows 7 Ultimate Edition WIM image that you have captured from a workstation on your SOHO network. You have used ImageX to install the image on another computer. What tool do you use to configure that computer to boot from the image?
- A.** BCDboot
  - B.** DISM
  - C.** BCDEdit
  - D.** ImageX

## Chapter Review

---

To further practice and reinforce the skills you learned in this chapter, you can perform the following tasks:

- Review the chapter summary.
- Review the list of key terms introduced in this chapter.
- Complete the case scenarios. These scenarios set up real-world situations involving the topics of this chapter and ask you to create a solution.
- Complete the suggested practices.
- Take a practice test.

## Chapter Summary

---

- You can use DISM or Image X to mount and unmount a system image. DISM adds packages, drivers, and updates to a mounted image and obtains information about online and offline-mounted system images. You can also use DISM to mount and service Windows PE images.
- MDT 2010 enables you to manage and manipulate disk images and to deploy them to target computers through a distribution share. Windows AIK needs to be installed before you can use MDT 2010 to deploy images.
- WDS creates a boot menu that you can use from a PXE-compliant computer booted from the network to install a system image to that computer. If a target computer is not PXE-compliant, you can boot it from a discover image to access the boot menu. A capture image is a type of boot image and appears on the boot menu. If you boot a reference computer from a capture image, you can capture its system image and copy it to a WDS server, which can in turn deploy it to other target computers.
- You need MDT 2010 (and Windows AIK), along with SCCM 2007 and SQL Server, to implement ZTI. You can implement LTI on PXE-compliant computers by using MDT 2010 and WDS.

## Key Terms

---

Do you know what these key terms mean? You can check your answers by looking up the terms in the glossary at the end of the book.

- **Boot image**
- **Commit**
- **Deploy**
- **Distribution share**

- **Install image**
- **Mount**

## Case Scenarios

---

In the following case scenarios, you apply what you've learned about deploying system images. You can find answers to these questions in the "Answers" section at the end of this book.

### Case Scenario 1: Deploying an Image with More Than One Language Pack

Don Hall, a systems administrator at Litware, Inc., has created a primary deployment image for Litware's client computers running Windows 7. He discovers that he needs to apply an update, add a new driver, change settings, and support multiple languages. He wants to make these changes without deploying the image and recapturing it. He also knows that if he services the image offline, he is not required to run the Sysprep tool and therefore does not need to use a rearm.

Litware is a relatively small organization with two locations. However, these two locations are in different countries. Don has created a single master image so that he does not have to maintain several variations. This image contains both the language packs that Litware requires. Don needs to apply updates to the single image and ensure that all the updates are applied to each language in the image. Before he deploys the image, he uses offline servicing to remove the unnecessary language pack. He needs to service only Litware's WIN image. There is no requirement to service a Windows PE image.

Don intends to use the DISM tool to mount and service the offline image and then unmount the image and commit his changes. He also wants to use DISM to create a report about the state of the drivers, applications, language settings, and packages that are installed.

1. What does Don need to carry out these tasks?
2. What is the first thing Don needs to do with the master image?
3. What tasks does Don carry out using the DISM tool?

### Case Scenario 2: Deploying an Image to 100 Client Computers

You are a network administrator at Northwind Traders. Your company is expanding its Detroit operation from a small branch office to a major facility, and you are tasked with deploying 100 client computers. The server infrastructure has already been upgraded. Detroit is part of the Northwind Traders AD DS domain, and DHCP and DNS servers are available on the Detroit subnet. The WDS server role has been installed and configured on a member server running Windows Server 2008 R2 and a standard boot image has been created.

You have been given a workstation running Windows 7 Enterprise that was previously used by a staff member that has left the company. All personal files and non-mission-critical applications have been stripped from the computer. Your boss wants all the new client computers to be installed “just like that one.” The target computers are new machines and easily meet the recommended Windows 7 Enterprise specifications. They are PXE-compliant.

1. What do you need to do to the workstation you have been given before you can use it as a reference computer?
2. What do you need to create in WDS to enable you to obtain an image from the reference computer?
3. How do you go about deploying the target computers?

## Suggested Practices

---

To help you master the exam objectives presented in this chapter, complete the following tasks.

### Manage and Manipulate a System Image

Perform this practice exercise when logged on to the Canberra computer with the Kim\_Akers user account.

- Use the DISM tool. Like all command-line tools, it might seem daunting to use at first, but it becomes familiar with practice. Mount the image on your bootable VHD, use all the Get options to obtain information, and then try all the configuration options. Commit your amendments to the image, boot from the VHD, and see the result of your changes.

### Become Familiar with the Deployment Tools

Perform the first practice exercise when logged on to the Canberra computer with the Kim\_Akers user account. The second and third exercises are optional.

- **Practice 1** MDT is already installed on your Canberra computer. Use the tool to install an operating system image, install drivers, install updates, install language packs, and install applications. Create and edit task sequences and associate answer files with the image.
- **Practice 2** Create a virtual PC running Windows 7. Deploy the system image that you created in MDT and observe the results.
- **Practice 3** Create a virtual server running Windows Server 2008. Install the AD DS server role and configure DHCP and DNS. Install the WDS role and practice using the Windows Deployment Services console and the WDSUTIL command-line tool.

## Take a Practice Test

---

The practice tests on this book's companion DVD offer many options. For example, you can test yourself on just one exam objective, or you can test yourself on all the 70-680 certification exam content. You can set up the test so that it closely simulates the experience of taking a certification exam, or you can set it up in study mode so that you can look at the correct answers and explanations after you answer each question.

### **MORE INFO PRACTICE TESTS**

For details about all the practice test options available, see the section entitled "How to Use the Practice Tests," in the Introduction to this book.

