Chapter 1

Planning the Deployment

Maintaining computers can be an expensive venture. In recent years, however, the cost of computer hardware has dropped to a drastically low level. Organizations have been able to leverage the cost drops and other volume purchasing programs to lower the initial expense of purchasing computers and Windows licenses. However, these initial purchase costs can pale in comparison to the cost of deploying the new computer systems.

Microsoft has provided many tools and capabilities with Windows Vista to help you reduce those deployment costs. Microsoft has redesigned its deployment process to provide faster and more consistent deployments. In addition, it has provided tools to customize and streamline the deployment process for your organization.

The focus of this chapter is to help you properly plan your deployment. There are many new technologies to master and many choices that must be made. If care is taken when making these choices, deploying Windows Vista can be an efficient process. Diving in without understanding some of these choices can ultimately lead to slow deployments, inconsistent desktops, project restarts, and time-consuming manual steps. This chapter aims to offer a starting point by providing an overview of key details and tools you should be aware of in order to get things off to a solid start including:

- Choosing the right edition of Windows Vista for your organization
- Determining the right method of installation
- Getting familiar with the Windows imaging format
- Choosing what should be included in your deployment image
- Automating the installation of additional applications following installation
- Addressing application compatibility issues

IN THIS CHAPTER

Selecting Windows Vista Editions

The most logical first choice in planning a Windows Vista deployment is to decide which edition or editions of Vista are to be used. With Windows 2000 and XP there was not much of a decision to be made — if you wanted to simply operate in a domain environment, Professional was the only choice. Vista makes this choice more complicated by offering several editions, but for most environments the choice will still be very clear. The key to making this decision is having a basic understanding of the differences.

For most organizations, only the Enterprise and Business editions will be a logical choice. That said, it is always good to be familiar with the real differences so you can make meaningful recommendations and defend any decisions made as to the edition to be deployed.

All editions support a maximum of 4GB of RAM on 32-bit systems. On 64-bit systems, Basic offers support for 8GB, Home Premium lets you work with 16GB, and the remaining Business, Enterprise, and Ultimate editions boast support for 128GB or more. With such a larger number of features available in the various editions of Windows Vista, it paints a clearer picture to state what you do not get with each edition. The list provides a quick summary of the features not included in each edition of Windows Vista:

- **Features not included with Windows Vista Ultimate:** None—that's why it's the ultimate.
- **Features not included with Windows Vista Enterprise:**
  - Parental controls
  - Windows Ultimate Extras
  - Themed slide shows
  - Windows Media Center (recording television, Xbox extensions, HD movie maker, and DVD Maker)
  - Small Business Resources
- **Features not included with Windows Vista Business:**
  - Parental controls
  - Windows Ultimate Extras
  - Themed slide shows
  - Windows Media Center (recording television, Xbox extensions, HD movie maker, and DVD Maker)
- **Features not included with the Windows Vista Home Basic and Windows Vista Home Premium editions:**
  - Support for two processors
  - Backup limitations including support for ShadowCopy or image-based system backup/recovery

- File system encryption
- Desktop deployment tools
- Policy based QoS networking
- Rights Management Services (RMS) Client
- Control over installation of device drivers
- Network Access Protection Client agent
- Pluggable logon authentication architecture
- Integrated smart card management
- BitLocker drive encryption support
- Support for worldwide interface languages or simultaneous installations of multiple user interface languages
- Subsystem for UNIX-based applications
- Virtual PC Express
- Windows Ultimate Extras
- Small Business Resources
- Windows fax and scan
- Wireless network provisioning
- Full support for Windows Mobility Center (thought it does provide partial support)
- Ability to join a network domain
- Remote desktop client support (though it cannot serve as host)
- Group policy support
- Offline files and folders support
- Client-side caching
- Support for roaming user profiles
- Support for folder redirection
- Ability to install IIS

- **In addition to the preceding items, the following additional features are also missing from Windows Vista Home Basic:**
  - Support for scheduled backups or for the backup of files to a network device
  - Aero user interface (glass, live thumbnails, dynamic windows, and so on)
  - Themed slide shows
  - Windows Media Center (recording television, Xbox extensions, HD Movie Maker, and DVD maker) Note: Home Basic does provide Windows Movie Maker (just not the HD version)

- Premium games
- It is limited to 5 SMB peer network connections (vice the 10 supported by the other editions)
- Tablet PC support
- Windows Slideshow feature
- Windows Meeting space support is limited to "view only"
- PC-to-PC synchronization
- Network projection
- Presentation settings

Vista Home Basic

Windows Vista Home Basic is the base code from which all other editions are built. It includes the new Windows Vista kernel and most security enhancements. Home Basic does not include the ability to join a domain. It also lacks most other features that would be useful mostly in business environments. Although this may make a suitable operating system for average home users, it has no place in a business environment.

Vista Home Premium

Windows Vista Home Premium includes all of the features of Home Basic and includes some additional features. One additional feature is the Aero interface, which gives us the glass-like interface and Flip 3D. On the more useful side, Home Premium includes support for tablet PCs, Windows Meeting Space, Scheduled Backup, DVD Maker, Windows Media Center, and additional games. Although these features make the operating system more fun and a little more useful, Windows Premium still lacks the ability to join a domain and other useful business features. In short, the Home editions should be used at home.

Vista Business

When using Windows Vista for business use, Windows Vista Business should be the first edition considered. Business Edition includes most all of the features of Home Basic but includes many additional features targeted at business customers. The following partial list of features that Vista Business contains makes it a more suitable choice in most organizations:

- Ability to join a domain
- Ability to apply Group Policies
- Remote Desktop
- Offline Files and Folders
- Tablet PC support
- Encrypting File System

- Complete and Scheduled Backup
- Windows Meeting Space
- Windows Fax and Scan
- Multiple physical processor support
- Volume, OEM, and Fully Packaged Product licensing options

Windows Vista Business also includes some more *nonbusiness* features, such as the Aero interface and additional games. In addition a very small number of home features, such as Parental Controls, are not available in Vista Business. This edition is targeted at general business use and is the edition of choice unless additional features of Vista Enterprise or Vista Ultimate are required. Though there are some features missing, such as BitLocker Drive Encryption, the features that are included make the Business Edition a good choice for fixed desktops and workstations (particularly since it is not likely that you will need full drive encryption for these systems).

NOTE **The features listed in this section simply document what is included in the editions of Windows Vista. This does not mean that you must install such features. The Windows Vista installation is customizable at a very granular level so that you may eliminate those elements of the setup you do not wish to include (games, for example).**

Vista Enterprise

Windows Vista Enterprise Edition is based on Vista Business, but includes features that some organizations may require or find useful. This edition is available exclusively to Microsoft Software Assurance customers, which may eliminate it as an option for some smaller companies. The features included in Enterprise Edition include all of those listed for Business Edition plus the following:

- BitLocker Drive Encryption
- Subsystem for UNIX-based Applications
- License includes the host and up to four virtual machines
- Ability to support multiple languages
- Volume licensing only

Depending upon your needs, any of these features may require you to move to the more expensive Enterprise Edition. Probably the most compelling feature of Enterprise Edition is the BitLocker Drive Encryption (also available in Vista Ultimate Edition) which makes this edition more suitable for portable systems. For test lab environments, the license to run four virtual machines without having to purchase additional licenses can actually lower the licensing costs of Enterprise to below those of Business. Before excluding Vista Enterprise from your options, be sure to consider the advantages and potential cost savings associated with the virtual machine licensing.

Software Assurance

If you are a Software Assurance customer, you may be entitled to some free licenses. For each Windows Client License covered under Software Assurance, you are entitled to one Windows Vista Enterprise upgrade license. The following Volume Licensing programs are eligible for this benefit:

- Open License
- Open Value
- Open Value Company-wide
- Open Value Subscription
- Select License
- Select License Software Assurance Membership
- Enterprise Agreement
- Enterprise Subscription Agreement

For more on the Microsoft Software Assurance program `see www.microsoft.com/licensing/sa/default.mspx`

Vista Ultimate

Windows Vista Ultimate is pretty much what it says. The Ultimate edition includes all the features from all of the other versions. It includes all of the features from the Home editions as well as the features from Business and Enterprise editions. Although this may sound like the best option for the organization desiring the best of the best, it has one characteristic that will exclude most organizations from using it. Like the Home editions of Vista, it is not available with volume licensing. The result will be that each computer must have a unique product key entered after installation, which somewhat offsets the advantages of automating deployment. Also consider that some of the features included in Vista Ultimate are simply unnecessary or undesirable in a business environment, such as Windows Media Center or Parental Controls. You could, of course, remove the features you don't want from the installation of Windows Vista Ultimate, but for the most part this would mean stripping the most expensive edition down to appear as Business or Enterprise. One feature that could be desirable for some organizations is Windows DVD Maker, but it is hard to argue this feature is worth the price and trouble. Consider third-party tools, such as offerings from Roxio or Ulead for such features, if desired.

Other options

In addition to the major editions above, Microsoft has also provided a few additional options. For example, Microsoft has provided a Windows Vista Starter edition for markets that are not classified as high income (high-income markets include the United States, Canada, the European Union, Australia, Japan, and New Zealand). It is a low-cost version of Vista which can only run a limited number of processes. This edition is not appropriate in a business environment and won't be available to most markets.

The other variant you may encounter are the N editions of Windows Vista. Due to legal issues in Europe, Microsoft also offers editions without Media Player included. These editions are identical to the standard editions above except the missing Media Player application and the addition of an N after the edition name. Unless your corporate policy requires an N edition of Windows, it is a simple process to remove Media Player from the installation or even block its use by using Group Policy.

Choosing a Deployment Type

The deployment of Windows Vista could be a great opportunity to establish a new and improved desktop. Others may feel they have their computers just as they should be. Your assessment of your current environment will likely be a key factor in deciding if an upgrade or a replacement is best for your organization. Other factors including the receipt of new computer hardware can also have an impact on how you plan your deployment of Windows Vista. Key deployment types covered here include:

- Replace
- Upgrade
- Refresh
- New computer

CROSS-REF **Key to a migration is the migration of user data and settings, which is detailed in Chapter 5**

Replacing computers

If you are replacing a computer, user data will need to be collected and stored in a temporary location. The new computer can be imaged with your customized image of Windows Vista beforehand in a staging area, but data restoral is typically performed on location (when the computer has been physically placed on a user's desk). It is a common scenario to replace only a fraction of the computers on a network on a rolling schedule. As such, a network often has new, old, and older systems which benefit from a cascading deployment. Power users get the newer (more powerful) computers, and their computers are reimaged and used to replace computers in the next tier of users. Finally, the oldest computers are expired or allocated to dedicated uses such as Internet kiosk stations.

Upgrading computers

As an alternative to migrating to Windows Vista, computers running Windows XP SP2 (or Windows Vista) may be upgraded to Windows Vista in-place. An upgrade retains your applications, files, and settings as they were in Windows XP SP2. Business and Ultimate editions of Windows Vista may only be applied as an update to Windows XP Professional or Windows XP Tablet PC. When moving from home editions, Windows 2000 and even Windows XP Professional x64 require a clean installation and cannot be upgraded in place.

Upgrades are typically discouraged as the introduction of a new operating system is an ideal chance to perform clean-up, employ lessons learned, and get a clean start. Regardless, applications need to be tested for compatibility. Performing an upgrade does not make incompatible software any less likely to exhibit issues.

CAUTION **The upgrade process will fail on target computers that have users logged on by using Remote Desktop sessions so be sure no such connections are active before initiating an upgrade.**

Refreshing computers

A refresh entails backing up user data and settings, installing a fresh Windows Vista image and then restoring user data and settings. Of course, this is a simple definition as there is much to be done including customizing the Vista image and addressing licensing and application needs.

Deploying new computers

Not so much a type of deployment, this scenario is in fact identical to that of a computer replacement. However, it is important to mention that new computers are often being shipped with OEM installations of Windows. So Windows Vista may well arrive installed and ready to go. However, unless you are a large organization with an arrangement with the manufacturer to provide a custom configured image, few corporate networks will accept the OEM installation as is. Often extra applications and promotional shortcuts are delivered with such systems. Therefore it is recommended that new computers arriving with Vista already installed be well scrutinized before accepting the provided image for use in your production environment.

Understanding Windows Vista Installation

When it comes to designing a deployment plan for Windows Vista, it is a good idea to first establish a basic understanding of its new deployment technologies and tools. If you are familiar with the deployment processes from previous Microsoft operating systems, you may be surprised by how many fundamental changes have been introduced with Windows Vista. Specifically, Vista now employs an image-based installation and leverages a detailed XML file for automation of the installation (as opposed to the simple INI file format used by previous versions).

Investigating the Windows Imaging format

Microsoft has significantly changed the installation process. Previously, numerous configuration screens were presented during installation, prompting for which components to install and other information to customize the installation. Using this information, extensive installation scripts were run to set up the initial environment. Installation times frequently ranged from 45 minutes to an hour for a standard installation. Microsoft has now moved to an image-based setup (IBS). This new image-based setup is based on Microsoft's new imaging format, the Windows Imaging format

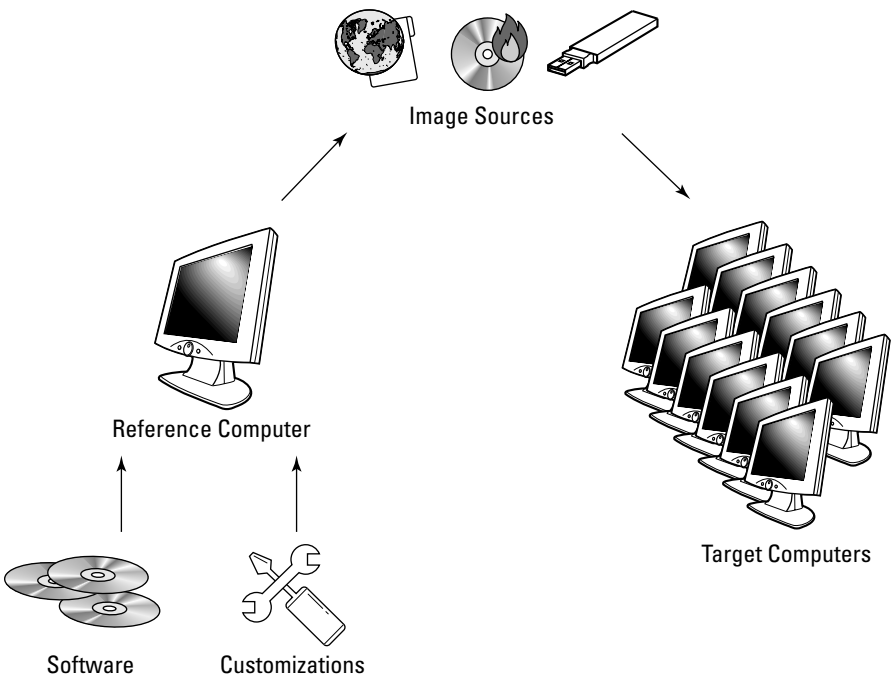
(WIM). Depending upon the computer and the customizations made to the installation, applying the image may take as little as 15 minutes.

Imaging is the process of making a copy of an ideal configuration and then replicating that copy to other computers. As shown in Figure 1.1, making an image involves installing an operating system on a reference computer, adding software and utilities, making customizations, and then making a copy of the reference computer. This image can then be applied to other computers to both speed deployment and ensure consistency throughout an organization.

For quite some time, Microsoft has been under pressure to better support imaging technology. Although Microsoft has not offered its own imaging solution until now, there have been several third-party vendors offering such solutions but with some inherent problems. In particular, there has been a need to provide better support of a single image on multiple hardware platforms and a unified imaging toolset. Providing better support of a single image allows organizations to support a much smaller number of images, ideally only one, that lowers support costs and increases consistency among deployed systems. Providing a unified toolset helps IT departments standardize imaging tools and reduce costs associated with third-party imaging tools. With the release of Windows Vista, Microsoft has attempted to address both of those needs.

FIGURE 1.1

Reviewing the imaging process



One limitation with imaging Windows has been the Hardware Abstraction Layer (HAL). Computers that use different HALs have typically required separate images. An example of computers that require different HALs is a single processor computer and a multiple processor computer, including hyper-threading and multi-core architectures. Another example of where HAL incompatibilities were often encountered was if one computer supported power management features and the other did not. Windows Vista now detects and installs the proper HAL, allowing you to use a single image.

Another factor that has forced many to use multiple images is supporting multiple mass storage controllers. Whether using SCSI, Parallel ATA, or Serial ATA, there are a large number of supported mass storage controllers available and most use a unique driver. For the most part, that still holds true. However, updating images to support new mass storage controllers has become much easier. Previously, extensive and often tedious answer file editing was required to update an image for new controllers. Now with only a few commands at the command prompt, your image can support the latest Windows Vista–capable mass storage controllers.

To address the problem of a unified imaging toolset, Microsoft has created the new Windows Imaging (WIM) format and several tools to manage and deploy WIM format images. The new format is file-based imaging rather than sector-based. Sector-based images are applied to the hard drive as raw data. The use of a file-based image format has several advantages.

One such advantage is that file-based images can be applied non-destructively. Microsoft is famous for their backwards compatibility and straightforward migration paths. An important migration path that must be supported for Windows Vista is an *in-place upgrade*. Although sector-based imaging is destructive and destroys all data on the partition to which it is applied, file-based imaging allows images to be applied leaving existing files in place.

Another advantage of file-based images is that they leverage the *single instance store concept*. Microsoft designed WIM files so that multiple images can be stored in a single WIM file. For example, a production image and a kiosk image can both be stored in the same file. Independently, this fact may seem rather esoteric. However, when combined with the fact that WIM files use single instance storage, this becomes a rather important advantage. If you were to compare two of your current desktop images, you would most likely find that a large majority of the files are the same, most likely over 90 percent. With single instance storage, only a single copy of each file is saved. This can drastically reduce image storage space. Microsoft has internally taken advantage of this technology by shipping one DVD with five separate editions of Windows Vista. This is possible because each of the editions shares a common code base, and only files not found in existing images are added to the WIM file for each additional image. Figure 1.2 illustrates the structure of a WIM.

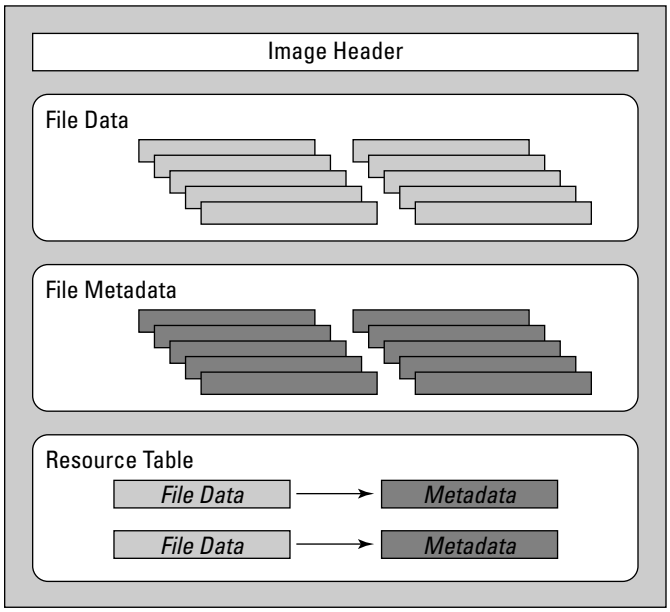
WIM file structure

WIM files begin with a header. The header of the file contains information, such as the compression type used and the signature and GUID of the image file. There are three types of supported compression. The first type, *no compression,* stores all files in their original state without attempting any compression. The image capture and apply operations tend to perform marginally better with no compression because the CPU is not required to compress or decompress each file during the

process. The second type of compression is referred to as *fast compression* and uses XPress compression. Finally, *LZX* compression, made famous by WinZip and PKZip, can be used to obtain high compression. Fast compression is the default method; it provides a good balance of CPU performance and space savings.

FIGURE 1.2

A Simplified diagram of Windows Imaging format file contents



NOTE **When applying images from a network location or slow optical drive, using no compression can provide worse performance than fast compression in many situations. This is true because the performance improvement of pulling a smaller image across the network often outweighs the extra work that must be performed by the CPU to decompress the data.**

The next section of a WIM file is the file data portion. In this section, the data from each file is first compressed and then stored. The other attributes, such as permissions and directory structure, are not saved here. As a matter of fact, the filename is not used here. Instead a hash is generated for each file. When the data hashes of two files are identical, only one copy of the data is stored, even if the filenames and permissions are different. This is known as a Single Instance Store (SIS) and is the largest section of most WIM files by far.

Immediately following the file data is the file metadata. Each entry in the file metadata section includes information, such as a filename, an access control list, and other file system attributes. Each entry maps directly to a file, except there is no actual data stored here.

All of the magic happens in the next section, the *resource table*, which is a table that maps file data to file metadata. The resource table also includes resource locations that are used to rebuild the directory structure. If you are familiar with databases, you may recognize this as a mapping table. If you are not familiar with databases, you can think of it as a large spreadsheet that connects the file data to its metadata.

WIM file distribution

After you understand how a WIM file works, you can address the issue of image application. Third-party imaging formats require third-party tools to apply those images to a hard drive. Applying a WIM file can be performed by using either Microsoft tools or a WIM-compatible third-party utility. A standard installation of Windows is performed with `setup.exe`, which replaces `winnt.exe` and `winnt32.exe`. Microsoft has also provided another tool, called ImageX, which can both create and apply WIM files.

CROSS-REF **ImageX is discussed in more detail in Chapter 6.**

Keep in mind, however, that WIM format images are file-based. File-based images do not store any partition or drive information. This offers the advantage of being very portable and being able to be used with varying-sized drives, assuming they can support the expanded data from the WIM file. File-based images do require, however, that the drive be prepared ahead of time. Disk partitioning and formatting must be performed before attempting to apply a WIM file image to a hard drive. This can be performed with any partitioning and formatting tool, but Microsoft provides the DiskPart utility to perform disk administration.

Third-party formats

Although this book focuses primarily on Microsoft's imaging format and tools, there are alternatives to consider (see Chapter 6). Microsoft has introduced its imaging format as the default distribution method, but that does not mean you are required to use it other than to initially install Windows Vista. Many organizations have significant time and money invested in other distribution solutions, such as Symantec Ghost and Acronis True Image. These alternative solutions often provide more than simple imaging features, so it is worth considering how you use them when deciding if such tools are truly replaced by the imaging tools now native to Windows. For some, combining Microsoft's tools with third-party tools may provide an efficient solution.

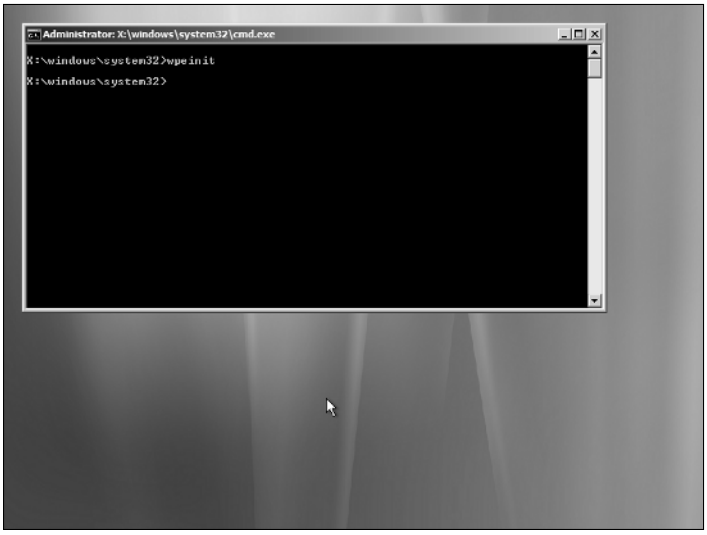
Leveraging Windows PE

Another major change to the deployment process is the manner in which installations are started. Before Windows Vista, Windows installation was typically started from DOS. DOS was used as the startup environment because of its small size and speed. Using such a small operating system allowed the installation to be launched from smaller media, such as floppy disks. DOS was also very familiar to systems administrators and fairly easy to customize and automate. Microsoft has now moved to a Windows-based installation environment, Windows PE. This section provides a simple overview of Windows PE; for more details see Chapter 6.

Windows Preinstallation Environment (PE) 2.0 is a scaled-down version of Windows Vista which uses a command prompt for the user interface, as shown in Figure 1.3. It does, however, have support for some very powerful features that make it very useful for both deployment and troubleshooting. Some of the included features are network support, Windows device driver support, Windows Scripting Host, and many other standard Windows tools and utilities. Utilities that aren't included can often be added or run from a network share. Due to the scaled-down features, Windows PE can be placed on a CD, DVD, or USB flash drive. A basic Windows PE implementation is less than 200MB in size.

FIGURE 1.3

Windows PE 2.0 user interface

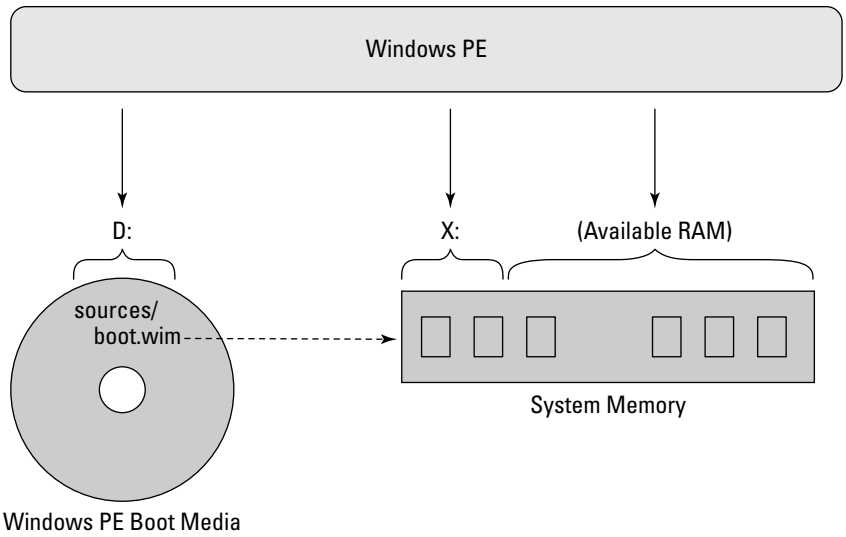


From a more technical standpoint, Windows PE is a Windows environment running entirely from memory. Successfully starting Windows PE requires two parts — boot files and a WIM image — to apply to a RAMDISK, as shown in Figure 1.4. During boot up, Windows PE boot media looks in a directory named sources for a file named `boot.wim`. Once found, `boot.wim` is imaged into a RAM Disk and given a drive designation of X. The portion of RAM designated as a RAM Disk is treated as a hard drive and is unavailable to the system for standard memory operations.

`Boot.wim` contains the entire Windows PE operating system. By default, the only other files present on the boot media are used to boot the system and image the `boot.wim` file into RAM Disk. By putting the operating system in RAM Disk, several things are accomplished. First, the boot media may be removed after boot because all necessary files are in RAM. Second, Windows PE requires read-write access to many of the system files, and RAM Disk is read-writable. This allows you to use read-only boot media. Using read-only media can be desirable, especially when dealing with virus outbreaks as the virus cannot infect read-only media.

FIGURE 1.4

An illustration of the Windows PE Architecture



Notice in Figure 1.4 that a hard drive is not required. However, if a hard drive is not present or partitioned, the boot media will be assigned drive letter C:. If hard drives are present and partitioned, the boot media will begin its lettering after all local disk drive letters have been assigned. The implication is that the boot media drive letter is unpredictable while the RAM Disk drive assignment is statically set to X:. If you plan to write custom scripts, keep this in mind.

Obtaining Windows PE has become much easier with Windows Vista. The distribution media uses Windows PE to perform Windows Vista installation. Microsoft has also made Windows PE, along with customization tools, available as a free download called the Windows Automated Installation Kit (WAIK).

As you begin to plan your deployment process, one of the first decisions you must make is what type of images to use. No matter what technology you choose, you'll need to be somewhat familiar with the native imaging technology as discussed in Chapter 6.

Although it may be tempting to base an image on a system already up and running with your corporate applications, doing so can cause trouble down the road. It is always advisable to create images based off of a clean operating system installation in order to attain a more stable and predictable baseline image. Document what you install and how you install it for accurate reproducibility. In fact, it is best to automate the creation of this image entirely by using answer files and automated application installations as discussed later in this chapter and throughout the book.

Default images

Windows installation times have been drastically reduced since the native installation method now utilizes an imaging process. For many smaller organizations, this increased efficiency in the installation process may be sufficient. If your organization only requires a couple of applications and doesn't typically re-image more than one or two computers a week, standard installations may work fine.

When using the generic installation image from the Windows Vista DVD, you may wish to automate the few remaining steps of setup. Customizations may include providing answers to the setup process and/or running post-installation scripts to install software. A couple of good examples of automated software installation are antivirus software and Microsoft Office suites, because it's quite likely that everyone in your organization will require these applications.

Custom images

Most likely, you will want to deploy your own customized images. Although Microsoft provides a generic image, it may require significant post-imaging processing to modify it to meet your organization's requirements. If you support over 500 workstations, imaging is probably a daily task and saving even a few minutes each time can be a huge advantage.

The real power of imaging is that it allows you to create a customized image unique to the needs of your company, not just what Microsoft thinks you might need. WIM files were designed with customizations in mind. In addition, free tools are available so that you can create, apply, and update WIM files. However, there are three basic types of images from which you must choose. These distinctions are less technical, and more philosophical. Before creating any custom images you must decide which philosophy your organization should follow.

Thin images

The philosophy of using thin images is one of flexibility. The idea is to create a plain image with very little, if any, additional software and customizations. All organizational software and customizations would then be added by scripts or a management solution, such as System Center Configuration Manager or Group Policy.

The advantage of a thin image is that it would require less maintenance because there is less software included to have to update and maintain. When software updates are required, the installation sources can simply be updated.

Thin images have also been used to reduce the number of supported deployment images. Because there is no software included on the image, a single image can generally be used to support many different workgroups, for example, Sales, Marketing, Production, and IT.

Although using a thin image eases administration, it slows down deployments. After applying the image, it may take a considerable amount of time for post-setup scripts to run and complete.

NOTE **For more information on working with thin system images see *Deploy Vista with a Thin System Image*, by Ruest and Ruest at: `http://itmanagement.earthweb.com/entdev/article.php/11070_3675806_1`.**

Streaming applications

An increasingly popular trend is the delivery of software as *virtual applications*. A virtual application is a way administrators may deliver software to users which isolates itself from the actual file and registry systems of the local computer. Chapter 4 discusses this in more detail, but virtual applications are often associated with another technology: *application streaming*. Application streaming allows for the delivery of software (typically virtual application packages) to computers over the network as they are requested by the user. Upon launching such an application, the required bits to start the application are downloaded to the client. As the application is running, additional bits are downloaded as necessary. Such data may also be cached so that it is only necessary to pull data from the network the first time the application is requested. While thin images may not seem practical, use of streaming technologies like this can make a thin image very compelling.

Thick images

Thick images are just the opposite. They are designed to include every possible piece of necessary software along with all desired customizations. The goal is speed and simplicity. After deploying a thick image, there is very little configuration to perform and no reliance on any external management software. Typically, naming the computer is all that is required.

The drawback of thick images is that the extensive customizations often make an image practical for only a small group of people. For example, including a Sales application in a thick image may preclude it from being used for the Human Resources department. The result is a larger number of supported images. Another drawback is keeping each application within each image up to date. The more applications you include in the image, the more updates and security patches you will have to apply.

Hybrid images

Most often, the ideal solution falls somewhere between thick and thin. These images are referred to as hybrid images. The goal is to support a minimum number of images while providing a relatively efficient deployment process.

A typical hybrid image would include customizations, such as:

- Microsoft Office suite
- Adobe Acrobat Reader
- Preconfigured Internet favorites
- Necessary desktop management agents
- Drivers for all supported hardware

Specialized or departmental applications are omitted to keep the image generic enough to be used throughout the organization. These applications would normally be installed by using a software distribution solution or post-imaging script, much like a thin image. However, by including software packages everyone will need in the image, the post-imaging processing times should be minimal.

Hybrid images are not without their problems. Because software packages are included in the image, it will require updating as each package is updated in your organization. However, hybrid images do strike a nice balance between performance and simplicity. This is the recommended image philosophy for most organizations.

When deciding just how "thick" or "thin" to make your images, ask yourself these questions:

- **Which applications are common to all workstations?** The more applications that all of your workstations have in common the more time can be saved by including them in the image. In addition, lengthy post-image scripts tend to be more error prone.
- **How long will it take to install each of the common applications?** It may make sense to include only the larger packages, such as Office suites. Often smaller packages, such as Acrobat Reader, can be installed quickly and more easily updated if excluded from the image. Keep in mind that you should only update the image if you update all of the existing machines in production.
- **Can the installation of the applications be automated?** If an application doesn't lend itself to automatic installation, it's certainly a good candidate to be included in the image.
- **What type of network connection will be available after imaging?** If you plan to image your machines with removable media in a remote location, thick images will require less network bandwidth. Properly designed, thick images will only require network access to join the domain, if applicable.

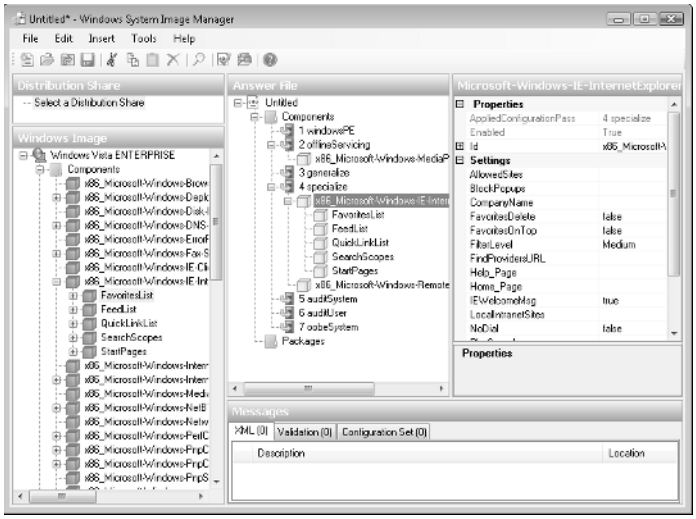
Automating the installation of Windows Vista

Building a corporate image is only the first step in planning your deployment. Although this image may contain the drivers and applications you want, its installation still requires the completion of a simple installation wizard. In previous versions of Windows, you could automate this process by dictating certain choices in your `Sysprep.inf file` or the `unattend.txt file`. While many administrators would edit this file directly, the key GUI tool for establishing these installation answer files was Setup Manager (`Setupmgr.exe`). The tool did not expose many of the customizations possible when editing directly, and with no error reporting or validation many administrators spent many hours fighting simple typos and confusing entries.

Like the installation format itself, the capability of automating the installation of Windows has also been drastically changed for the better. Today, System Image Manager (SIM) is the tool for not only creating these tools but also validating them to identify any problems that may exist prior to use. See Figure 1.5.

FIGURE 1.5

Reviewing the new System Image Manager tool



SIM is discussed in more detail in Chapter 2. The key thing to understand in the planning phase is the amount of power such a feature provides you and your deployment. Not only can you automate the installation options you see during an interactive installation, but there are also a great number of customizations you would otherwise never see.

A quick example of some things you can do by customizing an installation with SIM:

- Specify the owner and company names as well as the product key for the installation.
- Indicate whether to extend the partition to fill the remaining space on the disk.
- Customize IE home pages, control pop-up blocker, and dictate default favorites.
- Specify if users can use Remote Assistance to request help from a friend or support staff.
- Enable or disable the Windows Firewall.
- Dictate the name of the primary DNS domain to be searched for the name resolution.
- Specify the size and path of the page file.

TIP **Within SIM, you can right-click on any value and choose help to be provided with more details about each setting and value you may specify.**

CROSS-REF **You can use a freeware tool that goes by the name of vLite, which provides the ability to remove the things you do not want from the WIM image entirely. This tool is discussed in more detail in Chapter 6.**

Automating Application Installations

By using custom images, you can greatly reduce operating system deployment times. As discussed earlier, thick images can reduce deployment times but may very well require maintaining several images for your organization. For this reason, most organizations use a hybrid image. If using a hybrid or thin image, some or all applications must be installed after the imaging process. To create a more efficient and consistent deployment, the installation of these applications should be automated to the extent possible. This is not likely to be a new topic to you as it is common practice to automate installation. Further, this topic is covered in more detail in Chapter 4. This section provides a quick overview of some key software distribution techniques in use today to get you thinking about applications as part of your planned deployment.

Customizing application installation commands

The easiest option is often to use any automation or quiet installation support provided by the software vendor. Although it can sometimes be difficult to uncover them, many applications do provide command line arguments that can be passed to the installation program to force a silent installation. Depending upon the vendor you may be able to specify which options are installed and how much of the user interface is presented during installation. It is often this varying support for customized silent installations that can make automating application installations challenging.

Luckily, the Windows Admin community is a sharing one and many have shared details on the deployment of thousands of applications online at AppDeploy.com (`www.appdeploy.com`). This resource is a good starting point when working to identify which options exist. Although vendor Web sites often provide little to no information regarding deployment, what has been discovered or uncovered is organized here by application and version number.

The most ideal installation format to work with today is Windows Installer (MSI). As discussed in Chapter 4, Windows Installer offers a host of benefits including rollback, uninstall, logging, control of UI display, self-healing, nearly endless support for customization, and more. For applications to be certified as Windows Vista compatible, their installation package must be in the format of an MSI file. MSI files are small databases that instruct a service on your computer, the Windows Installer Service, what actions to perform to install, repair, or remove an application.

If the default installation options are acceptable, an application can often be installed using the /q option with Msiexec. An example of a quiet installation of Adobe Acrobat Reader with a progress bar during installation is

```
msiexec /i acroread.msi /qb
```

In most cases the MSI package and all supporting files can simply be copied off of the installation media. In some cases you must perform an administrative installation to a network share for silent installs to work properly. Two examples are Microsoft Office and VMware Workstation. As is the case for many applications distributed via the Internet, Adobe Acrobat Reader is actually a self-extracting archive that decompresses and then runs a Windows Installer setup. To get at these

source installation files, start the setup, and when the Welcome screen appears, look in the `%TEMP%` folder for a new subdirectory (often with a random name), which contains the installation source. Copy these files to your network share and run such setups directly.

InstallShield (the most common ISV setup type) supports the ability to pass MSI command line parameters from the `setup.exe` so that it need not be extracted (or pulled from `%TEMP%`) if you do not wish. However many InstallShield setups are actually programmed to check if they are being run from the provided `setup.exe`. The reason is that some InstallShield setups rely upon an installed setup scripting component that must be installed prior to running the intended setup. Such setups require that you not call the MSI setup directly, which can be frustrating to administrators. Fortunately, you can use a couple of workarounds, including passing the property ISSETUPDRIVEN to the command line installation and assign it a value of 1. For example:

```
msiexec /i mysetup.msi ISSETUPDRIVEN=1 /qb
```

NOTE **A video presentation on working around such setups may be found online at `www.appdeploy.com/video/installscript.asp`.**

Creating transform files

When simply instructing an application to install silently is not sufficient, transform files provide you the ability to inject customizations. Microsoft Transform (MST) files can act as answer files that can be passed to the Windows Installer Service to select components to install and set preferences. Aside from choosing options, you can make any change you want to the provided setup including adding and removing files and registry entries. Some vendors even provide tools to help you create an MST file. For example, Microsoft Office MST files can be created with the Custom Installation Wizard.

However, most applications do not offer their own tool for generating an MST file. For such applications, you can rely on a third-party tool to customize the installation this way. If you are looking to make simple changes, such as adding or changing property values, it may suffice to make use of the free ORCA Windows Installer table editor from the Windows Installer SDK.

NOTE **The Windows Installer SDK is available free from Microsoft at `www.microsoft.com/downloads/details.aspx?FamilyId=A55B6B43-E24F-4EA3-A93E-40C0EC4F68E5&displaylang=en`. Once installed, you can find the installation at `<InstallationDirecotry>\Microsoft SDK\bin\Orca.msi`.**

After an MST file has been created, it is passed as an option to the `msiexec` command to perform a silent installation. For example

```
msiexec /i mysetup.msi TRANSFORMS=mytransform.mst /qb
```

NOTE **If deploying a Windows Installer setup via Group Policy, you can specify one or more transforms as options in the GPO. If you desire to pass a property to an MSI setup being deployed by Group Policy, you may use a tool such as ORCA to specify the value, as command lines may not be specified in Group Policy.**

Repackaging applications

If your application does not support a silent installation or the necessary level of customization can't be performed through silent installation, you need to repackage your application. Repackaging requires the use of third-party software (ORCA does not offer a repackaging feature). The two most widely used repackaging tools are AdminStudio from Acresso Software and Wise Package Studio from Symantec.

Repackaging tools typically work either by monitoring the changes made by a setup or by means of a *snapshot* that detects the differences made to a system by a setup. A snapshot has the user begin with a clean workstation that contains a minimal amount of additional software. The repackaging tool then takes a snapshot of the workstation before and after the installation of an application and determines those detected differences to be the needed contents for the newly created MSI setup.

Although MSI is the most common target for a repackaged application, application virtualization is developing as a popular alternative. Virtual applications run in memory and do not update the local computer's file system. While the act of creating a virtual application is often similar (or identical) to that of repackaging a setup to MSI format, the resulting package is instead managed by a client agent (or even its own internal mechanism) as opposed to being managed by Windows Installer. Repackaging a setup into a virtual application format is sometimes referred to as *sequencing* (a term coined by SoftGrid).

NOTE **A detailed list of applications that offer repackaging functions may be found online at `www.appdeploy.com/techhomes/windowsinstaller.asp`.**

Selecting a Distribution Media

After you have determined if a thick or thin image will be used, the required distribution media may be chosen. Options include network, DVD, or USB drive distribution. Although variances and combinations may be used, these are your basic choices.

DVD

DVD distribution has several advantages if all of your computers have DVD-ROM drives. DVD media is relatively cheap, and there is little upfront cost because most modern computers do have DVD drives. DVDs are also large enough to handle modestly sized images. The generic image from Microsoft is about 1.8GB. Therefore, you will be able to add several applications and still stay within the size constraints of the DVD. When you take into consideration including Windows PE on the DVD, you have about 4GB available on a DVD with which to consider adding applications. Keep in mind that even if you have to use multiple images in your environment, WIM files allow multiple images to store a single instance of each file. Therefore it is very practical to store two or three images on the same DVD.

Like everything else, DVDs do have drawbacks. Three items that most often exclude DVD as a distribution choice are size, reliability, and maintenance. If your images turn out to be larger than 4GB, DVDs become very inconvenient. You may span an image across several DVDs, but that means an additional manual intervention. Reliability also comes into play, because DVDs are easily scratched, particularly those burned rather than pressed at the factory. Reaching the end of the last DVD of a three-DVD set and having the installation fail due to a damaged DVD can be very frustrating. Most problematic is the issue of version control. When your deployment image is updated, you must create new DVDs and distribute them to every technician responsible for imaging. If a single technician continues to use an older image, the network can be put into an inconsistent and potentially unstable state. This is particularly easy to see if you are moving to a new Windows service pack or Office Suite version.

USB drives

The newest edition to the list of distribution media is USB drives. With the inclusion of bootable and high-speed USB support on nearly every new computer, USB drives become a very attractive distribution media for some environments.

USB drives have several advantages over DVDs for distribution of your deployment images. One reason is that USB drives are typically much faster than DVD drives. A fast USB drive can boot and apply an image in less than half the time it might take to apply the same image from DVD. Secondly, they require no network connectivity and can be useful in remote locations or on overworked networks where bandwidth is a concern. Also, USB drives tend to be more reliable and are not susceptible to scratching in the way DVDs are. Lastly, the typical USB drive is much larger than the capacity of a DVD, allowing for more and larger images on a single drive.

USB drives suffer some of the same downfalls that DVDs do, namely not having a reliable means of version control. When new versions of the deployment image or images are created, every USB drive must be updated with the new image or images. Although the reliability of USB drives is higher than DVDs, they are still vulnerable to being dropped or broken.

NOTE **If you are using DVDs or USB drives in a networked environment, consider including a simple script that checks a network share for a simple file containing the latest build number. This way the convenience of removable media may be enjoyed while minimizing the concern of version control.**

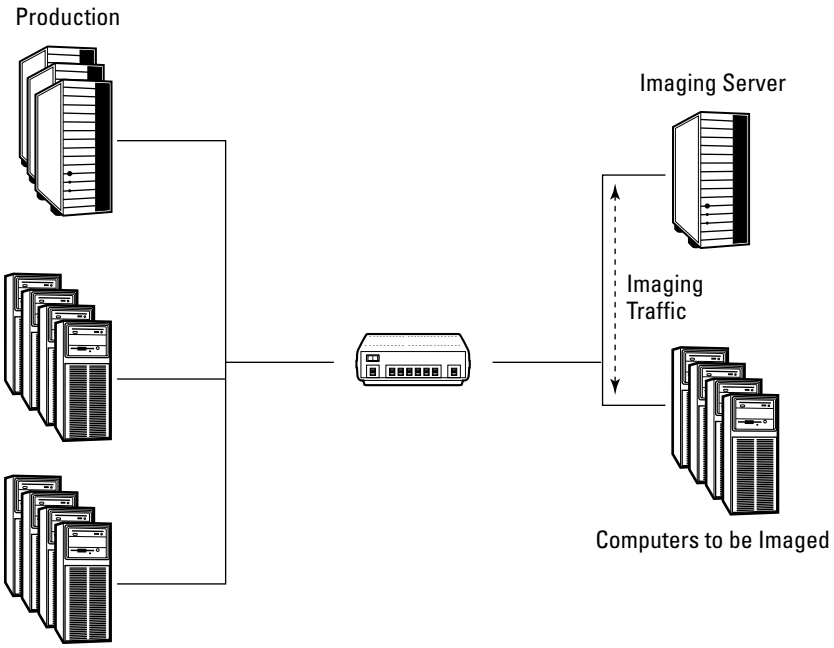
Network

In environments that enjoy high-capacity and efficient networks, network distribution is recommended when possible. Using the network to distribute your image offers the advantages of centralized version control and, hopefully, reliability. Centralized version control is realized because all imaging is performed from the same distribution point. When new images are released, a simple update to the distribution point ensures that all new imaging will use the new image. Reliability is also increased as networks cannot be scratched or dropped as can DVDs or USB drives.

On the downside, your network must be able to handle the additional load of pulling deployment images across them. An unreliable or slow network can become even more so once gigabytes of information are being transferred for each image. If your network supports the ability to multicast, network impact can be minimized greatly by sending the data over the network only once with several target computers listening to receive it as a single stream. Another potential solution to minimize the impact of network-based imaging is to perform all of your imaging on an isolated or segregated network. In larger environments, a dedicated imaging server and network switch are often employed on a separate subnet from the production environment, which may also be used as a staging area for the introduction of new computer hardware, as shown in Figure 1.6.

FIGURE 1.6

Sample network configuration for imaging



Evaluating Hardware Requirements

If you will be migrating existing machines to Windows Vista, you will first need to get an idea of what hardware you have deployed and how well it will handle Windows Vista. With its introduction, much hype and media attention was given to how much more hardware intensive Windows Vista is. In reality, Windows Vista is quite an adaptive operating system and will adjust its features to accommodate the hardware on which it is installed. However, there are some minimums that must be abided by and some recommendations to keep your users from becoming unduly frustrated at sluggish performance.

If you have an inventory system in place, you can do some quick checking against the requirements to identify those in need of updates. But even with a third-party inventory tool in place, it is advisable to check each machine representative of your environment for hardware compatibility using the Windows Vista Upgrade Advisor. More than a simple requirements check, this tool interrogates the local computer hardware and checks for compatibility using the latest information from Microsoft's Web site. If there are just a handful of different hardware types in your environment, running this on each of them can provide some valuable data.

Another option is the new Microsoft Assessment and Planning Solution Accelerator (MAP). This inventory, assessment, and reporting tool be used without requiring the installation of agent software on any computers or devices. While similar solutions often require installation of a client agent to collect local details, MAP users Windows Management Instrumentation (WMI), the Remote Registry Service, SNMP, Active Directory Domain Services, and the Computer Browser service to collect this information without a need to deploy remote agents. It provides identification of currently installed Windows Client operating systems, their hardware, and recommendations for migration to Windows Vista. In a medium to large environment this offering replaces the need for the Windows Vista Upgrade Advisor.

NOTE **The Windows Vista Upgrade Advisor is available online at `www.microsoft.com/windows/products/windowsvista/buyorupgrade/upgradeadvisor.mspx` and the Microsoft Assessment and Planning Solution Accelerator is available online at `www.microsoft.com/downloads/details.aspx?FamilyID=67240b76-3148-4e49-943d-4d9ea7f77730&DisplayLang=en`.**

As far as minimum requirements go, Microsoft has published charts outlining the bare minimums for use of Windows Vista as well as the recommended minimums to ensure support for full functionality (particularly to support the Aero interface). Table 1.1 outlines the recommended hardware requirements from Microsoft as they apply to the Business, Ultimate, and Home Premium editions of Vista.

TABLE 1.1

Recommended Hardware Requirements

Processor	1 GHz 32-bit or 64-bit processor
Memory	1GB of system memory
Hard drive	40GB (15GB free)
Video	WDDM driver 128MB Video Memory Pixel Shader 2.0 in Hardware 32 bits per pixel
Media	DVD-ROM drive
Sound	Audio Output

With these specifications, your computer may be able to perform basic tasks, such as working with e-mail and surfing the Internet, but the performance of more demanding applications will certainly suffer. In general, consider the following items when evaluating whether your current hardware should be reused, upgraded, or replaced:

- **Does my hardware meet the minimum requirements**? If your hardware doesn't even meet the manufacturer's recommended minimums, don't even try it!
- **What type of applications and processing will be performed on this computer?** Quite often computers are used for little more than e-mail and Internet access. Relatively humble hardware may do the job just fine. For other applications, always check with the application vendor for their recommended minimums.
- **How reliable is the existing hardware?** Typically, after four or five years, hardware tends to become less reliable, especially laptops. If hardware is unreliable it will cost the organization much more in IT support costs, repair costs, and end user downtime.
- **Will it be more economical to upgrade or replace?** If you decide that your hardware will not be sufficient for the tasks at hand, you must evaluate whether an upgrade or replacement will be more economical. Be sure to include factors such as warranties and time involved in performing upgrades versus fresh installs.

Also keep in mind the option of thin clients. If your existing hardware meets all of your needs with the exception of having an application or two, then using the old hardware as a thin client may extend the life of the hardware by several years. It may also help reduce management costs and have other advantages.

The most common hurdle during this part of deployment is to obtain an inventory of existing hardware. If your organization has an inventory system already in place, a simple report should give you the information you need to begin your planning. If not, there are several free options to get an inventory of your existing hardware.

The most flexible solution is to use some type of scripting to obtain the necessary information. Windows Vista supports scripting through the command line (batch or shell), Windows Scripting Host, and PowerShell. The great thing about creating your own script is that you can customize it to provide the data you need in the format you want.

If you don't already have an inventory system, there are several free tools, such as Open-AudIT (`winventory.sourceforge.net`) and Lansweeper (`www.lansweeper.com`) to help you get an inventory of your deployed systems. A search at `www.sourceforge.net` for hardware inventory returns numerous results of projects and applications to help inventory your network. Likewise, a search on Google for free hardware inventory returns several products to aid with the task. Ultimately, you will need to find a product that meets your corporate standards and provides you with the information you need.

ON the CD-ROM **Lansweeper is included on the CD accompanying this book.**

Ensuring Application Compatibility

Microsoft has introduced many new features with Windows Vista, and many of those focus on security. One of the problems with increasing security is that it typically implies a decrease in ease of use. One very visible example of this is application compatibility. If you've been in IT long enough to have been through an operating system migration, you know the potential for pains associated with application compatibility. When planning your deployment, you must take into account the compatibility of existing applications and the possibility of replacing or upgrading incompatible applications.

Understanding broken applications

Applications that have previously functioned correctly may now cease to function in Vista for many reasons. Microsoft has encouraged developers to write applications that run properly for standard users for some time, but now more implicitly enforces this as a requirement. When users must have Administrative rights to execute an application, it makes the system more vulnerable to accidental damage, virus attacks, and malware. With the strengthened file system and registry permissions, standard users are not generally able to damage a system to the point where it won't boot, nor are they generally allowed to perform actions required for most virus activity. Although it can initially seem easier to simply grant users administrative rights, doing so can cause a host of new problems by the unknowing (or even intentional) damage that may be done by users. With Windows Vista, simply giving users admin rights is much less of an option in that even when logged on as an administrator processes are not run with those elevated privileges without being prompted.

The prompting of the system for confirmation to perform administrative tasks (or changes that affect more than just the local user of the computer) is known as User Account Control (UAC). When an administrator logs into Vista, the shell is started without the administrative token. The result is that any applications are run as a standard user. When performing a task that requires elevated privileges, the system prompts to alert the administrator as to the seriousness of their actions. It will also prompt standard users, except they must enter account credentials for an administrative user, because they themselves don't have permissions to perform the action.

Of course, as with any prompt that pops up very often, users will begin approving them without reading the warning. Although effective, it is a rather unpopular feature and many have taken to disabling it rather than work around any issues. Obviously, the recommendation is to keep it in place and work around any issues encountered.

In some cases, the problem is not security related but is truly an application compatibility problem. One possible issue may be if an application requests direct hardware access. Because making a direct hardware request has not been allowed since Windows ME, this situation not very common. More common is that the operating system answers requests from applications in a slightly different fashion than in previous versions. This is due to changes in system files and the way in which API calls must be made. The good news is that these problems are probably the easiest to solve.

NOTE **Application developers must be aware of new requirements and best practices when it comes to application security on Windows Vista. Particularly if you have an in-house development team, some useful reading to recommend would be:**

Developer Best Practices and Guidelines for Applications in a Least Privileged Environment

`msdn.microsoft.com/library/default.asp?url=/library/en-us/dnlong/html/AccProtVista.asp`

The Windows Vista Developer Story: Application Compatibility Cookbook

`msdn.microsoft.com/library/default.asp?url=/library/en-us/dnlong/html/AppComp.asp`

Microsoft Standard User Analyzer included in Microsoft Application Compatibility Toolkit Version 5.0

`technet2.microsoft.com/WindowsVista/en/library/1082691c-8f61-44a8-a55f-000c0b80e10f1033.mspx?mfr=true`

Identifying application incompatibilities

If your application is not functioning correctly under Windows Vista, check with the vendor or look for information on their Web site. Often application incompatibilities are known and patches or updates are released. If the vendor is unwilling or unable to correct the application, then you may need to proceed with correcting the problem yourself.

Identifying permission issues

You should begin by identifying whether the problem is permissions based. The easiest way to test this is to launch the application by right-clicking its shortcut and selecting Run as administrator. If the application functions correctly, you are most likely dealing with a permissions issue. These can usually be addressed by loosening security on a folder or registry key. Although this does slightly decrease the security of the system, it is far better than giving users administrative rights.

There are a few ways to identify permission issues with an application. One way is to use the Process Monitor utility; another is to enable auditing of object access failures, and yet another is to try Application Rights Auditor from BeyondTrust.

NOTE **There are a couple of commercial tools available to detect permission issues on the fly such as BeyondTrust Privilege Manager (`www.beyondtrust.com/products/PrivilegeManager.aspx`) and Symantec (Altiris) Application Control Solution (`www.altiris.com/Products/AppControlSol.aspx`).**

Using Process Monitor

Process Monitor has been around for some time as a freeware tool from SysInternals. The well-known File Monitor (FileMon) and Registry Monitor (RegMon) tools have been superseded by Process Monitor, which is now the tool to use for monitoring both the file and registry systems. The following steps illustrate how to use Process Monitor to identify where permission problems may be causing an application to fail.

1. **Start by downloading Process Monitor from Microsoft's Web site:** `http://technet.microsoft.com/en-us/sysinternals/bb896645.aspx`
2. **Extract the `Procmon.exe` utility from the downloaded archive and execute it (you will receive a UAC security prompt).** Process Monitor will immediately start recording everything the computer is doing.
   - To stop it, choose File ➪ Capture Events (or click Ctrl+E).
   - To clear the display choose Edit ➪ Clear Display (or hit Ctrl+X).
3. **Prepare to reproduce your problem; when ready, repeat the previous step to restart the monitoring process.**
4. **After you have reproduced your error, stop the capturing of events and review the information recorded.** Any file or registry permissions that may need to be modified should be recorded with a result of ACCESS DENIED. Double-click each such listing for full details.

Using audit policies

If permission issues are the cause of application compatibility problems, you may enable auditing to help locate the resources that are attempting to be accessed. Configure auditing to locate the problem by following these steps:

1. **Open the Group Policy Object Editor by running gpedit.msc from a command prompt or Start Search box.**
2. **Navigate to Local Computer Policy ➪ Computer Configuration ➪ Windows Settings ➪ Security Settings ➪ Local Policies ➪ Audit Policy and double-click Audit object access.**
3. **Select the failure box to enable failure auditing. Click OK and close the Group Policy Object Editor.**
4. **Open the Properties window of the system drive, typically C:, and select the Security tab.**
5. **Click the Advanced button, change to the Auditing tab, and click the Continue button.**
6. **When presented with the Auditing interface, click Add and type Everyone as the account to audit. Click OK to continue.**
7. **Select the Failed box beside Full control to enable failure auditing for all events. Click OK until you exit the security window.** You will get several errors that auditing can't be enabled for some folders and files. In general, these files and folders are not problematic and can be ignored.
8. **Open the registry editor,** `regedit.exe`**. Right-click on** `HKEY_LOCAL_MACHINE` **and select permissions**.
9. **Use the previous steps used on the file system to enable failure auditing for everyone and close the registry editor.**
10. **Run the application once again, and once it fails, review the Security log with Event Viewer.** You should see failure audits for folders or registry keys that the application failed to access or change.

After the file, folder, or registry key causing the problem has been located, it can be adjusted. Typically the files and registry keys causing the problem are beneath a vendor or application directory or key. Don't forget to reverse the procedures above to disable auditing after the process is complete. Otherwise, unnecessary auditing may slow down your system substantially.

Using BeyondTrust Application Rights Auditor

One (free) tool available is an Application Rights Auditor — a free product that automatically identifies and reports which Windows applications require users to have administrative rights. Application Rights Auditor uses a Microsoft Management Console (MMC) snap-in interface in conjunction with a desktop agent. The desktop agent is installed on computers to examine applications during execution. All application executions are automatically monitored silently in the background, and it sends pertinent data to a server where centralized reports may be configured and viewed within the MMC.

Application Rights Auditor is available from `www.beyondtrust.com/products/ApplicationRightsAuditor.aspx`

Addressing application incompatibilities

Depending on the compatibility issue you have identified, there may be one or more different approaches to realizing a solution. This section provides some basic guidance on addressing both application security issues as well as programmatic incompatibilities.

Modifying application security

After you have identified areas of the system that require permission changes in order to support needed applications, you will have several options for deciding how to implement those changes. Particularly if you determine the need before rolling out Vista, you have the option of including the permission changes in your custom image. Some choose to include such permission changes along with the automated installation package using Windows Installer or scripts. There is no correct answer; it really depends upon your environment and with what you are comfortable. The two most common methods for changing security are Group Policy and the XCacls command line utility.

Using Group Policy

Particularly if you are using Group Policy to deploy a Windows Installer setup already, enforcing required permission changes in the same Group Policy Object is ideal. You can create separate GPOs for permission changes like these, but consider the future when planning your strategy. If you later remove or upgrade an application from your environment, will you be able to easily remove those security changes along with it? Some choose to keep all file and registry permission changes in a single GPO for performance reasons. This is not a bad idea, but be sure to carefully document each change so you can easily associate it with the appropriate application in the future.

To apply or modify permission entries for objects using Group Policy:

1. **Open Microsoft Management Console (`mmc.exe`) and choose File ➪ Add/Remove Snap-in and then select Add.**
2. **Select Group Policy Object Editor and click the Add button.**
3. **In the Group Policy Wizard, on the Select Group Policy Object page, click Browse and either select an existing Group Policy object in the appropriate domain, site, or organizational unit, click OK, and then click the Finish button.** (If you want to create a new GPO, right-click and choose new in the desired location.)
4. **Click the Close button and then click OK to complete the adding of the snap-in.**
5. **To edit the security of a file or folder navigate to Computer Configuration Windows Settings ➪ Security Settings ➪ File System. To edit the security of a registry key navigate to Computer Configuration ➪ Windows Settings ➪ Security Settings ➪ Registry and click Add for the appropriate item.**
6. **To set permission for a group or user either select (it if it is already listed) or click the Add button (to specify an unlisted one). To allow permission, in the Permissions for User or Group box, select the Allow check box. To deny permission, in the Permissions for User or Group box, select the Deny check box.**

Using XCacls

XCacls is an updated version of the included cacls command line tool and offers much better command line support. XCacls is available online at `http://support.microsoft.com/kb/318754`. It offers several command line arguments to dictate its action (as shown in Table 1.2). Its syntax is as follows:

```
xcacls FileName [/T] [/E] [/C] [/G user:perm;spec] [/R user] [/P
   user:perm;spec [...]] [/D user [...]] [/Y]
```

TABLE 1.2

XCacls Command Line Arguments

Parameter	Description / Options
Filename	The file or folder to which permission should be applied (wildcards are supported)
/T	Recurse through the current folder and subfolders
/E	Edit the permissions instead of replacing them
/C	Continue if an Access Denied error message is encountered
/G	Grants the specified user the specified permissions in the following format: User: perm;spec **User**: The user or group name to be granted permissions **Perm**: The permissions to be granted **C**: Change (write) **F**: Full Control **P**: Change Permissions (special access) **O**: Take Ownership (special access) **X**: Execute (special access) **E**: Read (special access) **W**: Write (special access) **D**: Delete (special access) **Spec**: Applies only to folders and accepts the same values as perm, with the addition of the following: **T**: Not Specified. Sets the permissions on the directory without specifying permissions that are to be applied to new files created i n that directory. At least one access right has to follow.
/R	Revokes all access rights for the specified user

continued

TABLE 1.2 *(continued)*

Parameter	Description / Options
/P	Replaces access rights for the specified user in the following format: User: perm;spec **User**: The user or group name to be granted permissions **Perm**: The permissions to be granted **C**: Change (write) **F**: Full Control **P**: Change Permissions (special access) **O**: Take Ownership (special access) **X**: Execute (special access) **E**: Read (special access) **W**: Write (special access) **D**: Delete (special access) **Spec**: Applies only to folders and accepts the same values as perm, with the addition of the following: **T**: Not Specified. Sets the permissions on the directory without specifying permissions that are to be applied to new files created in that directory. At least one access right has to follow.
/D	Denies the user access to the specified file or folder
/Y	Disables confirmation prompt when replacing user access rights

An example of using the XCacls command line utility is provided here:

```
xcacls.exe "%ProgramFiles%\Application\settings.xml" /E /G
   "Authenticated Users":C
```

This example uses the /E switch to instruct the tool to edit and not replace the security settings for the specified file. The /G switch says to grant rights to the specified group Authenticated Users and the C specifies that it is to grant them Change permission.

Investigating compatibility modes

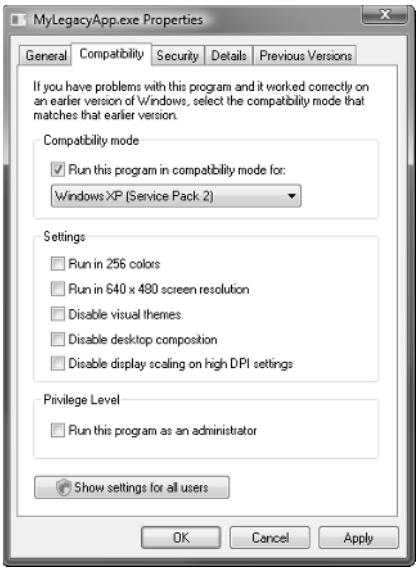
Some applications do not fail for lack of permissions. These applications often fail because of an unexpected response from an API call to Windows. One such situation is version checking during installation. Some older installation programs check to ensure that a particular version of Windows is present, for example Windows 98. When the system reports that it is running Windows Vista, the installer fails, stating that it requires Windows 98 or newer. Obviously Vista falls under *or newer*, but the install application simply doesn't recognize the new name.

A multitude of other examples and situations exist, but it all comes down to what to do to fix the problem. If running your application as an administrator doesn't indicate a permissions problem, next try setting a compatibility mode for the application.

Technically speaking, compatibility modes tell the operating system to respond to application calls as if it were a previous version of Windows. In addition, when the operating system is asked to identify itself to the application, it lies depending upon which compatibility mode has been configured. An example is shown in Figure 1.7 of setting a legacy application to run in Windows XP compatibility mode. You may also choose from Windows 95, 98, ME, NT 4, 2000, and Server 2003.

FIGURE 1.7

Viewing application compatibility options



Depending upon the application, you may also need to adjust the visual settings. If a visual setting must be adjusted it is usually easy to recognize. Symptoms may include windows that don't resize properly, display improperly, or give an error stating they must be run with 256 colors. Do not use this setting unless necessary; as when launching an application with visual compatibility mode settings, the system has to make visual settings changes that sometimes cause problems with other running applications that need to be run with regular Vista display settings.

When selecting a compatibility mode, always start with the least intrusive. That is to say start with XP or Server 2003 compatibility modes first because Vista must only change a few responses to emulate those relatively new operating systems. Windows 95 compatibility mode requires that the system change a lot about the way it deals with the application, and changing too much can both slow down the application and possibly cause new problems due to unnecessary legacy emulations.

Aside from right-clicking on an executable and setting these options from the compatibility tab of the properties dialog, there is also a Program Compatibility Wizard which may be used. To access the Program Compatibility Wizard click the Start button, then Control Panel, then Programs, and then click Use an older program with this version of Windows.

Finally, there is the Program Compatibility Assistant (PCA) which is an automated feature that attempts to detect and apply solutions to known compatibility issues in an automated manner. A common scenario where you may see the PCA feature present itself is in a setup scenario. Some cases where PCA may assume a setup may have had issues include:

- No Add/Remove Programs (ARP) entry is created by the installation
- An Access Denied error level is returned by the setup executable when exiting
- UAC does not detect the executable as a setup so that the opportunity to execute as administrator is presented
- The executable uses the GetVersion function or the GetVersionEx APIs to get information on the Windows OS version and fails when hard-coded to look for version 5 (Windows XP)

When PCA detects these scenarios it will prompt the user to run the setup again with recommended settings. Then, based on the scenario PCA will run the setup and attempt to mitigate the detected issue by automatically applying the correct solution (running as Administrator, applying version compatibility settings, and so forth).

NOTE **This is an automated feature and not a wizard or tool that can be executed manually. For details about the types of issues PCA addresses, and how it mitigates the problem when re-running the installation, see `msdn.microsoft.com/en-us/library/bb756937.aspx`.**

Application Compatibility Toolkit

If your organization has been using Vista for some time, then only a small set of applications may need to be tested as the rest may have already been field tested. However, if you are migrating to Windows Vista, testing all of your critical organizational applications can be quite time consuming and costly. Because of time and cost, secondary applications are often left to be tested by the end users. This can lead to downtime and a general dislike for the IT department. Microsoft has introduced a free toolkit to help aid with inventorying and testing application compatibility, the Application Compatibility Toolkit.

NOTE **Download the Application Compatibility Toolkit online at `www.microsoft.com/downloads/details.aspx?FamilyID=24da89e9-b581-47b0-b45e-492dd6da2971`**

The Application Compatibility Toolkit (ACT) has been designed as a four-stage solution for addressing compatibility problems. First, it helps you gather information about your existing applications. Second, it can analyze existing applications to recognize potential compatibility problems. Third, it has the ability to create fixes for the problems, which you can then distribute to the appropriate computers. Finally, it provides the ability to share information with others via the Microsoft Compatibility Exchange.

To begin the process, you need to identify a computer to host the data collection process. Ideally, you would use a server and that server would be dedicated to the deployment process. The collection computer must have ACT installed as well as an available SQL server. If you have a production SQL server, ACT will create a new database for collection. If you prefer, you can install Microsoft SQL Server or even Microsoft SQL Server Express (free) on the collection computer to reduce the impact on production.

After installation, a collection plan is designed and ACT builds an executable that can be executed on target computers. This will collect application data and log it to the collection computer. It may take several days or weeks to get a complete list because some users may be out of the office temporarily. The ACT collection package is an MSI setup that requires administrative rights so you can deploy it via Group Policy, manually on a computer (as Administrator), or, if you have a systems management solution in place, you can also deploy the package as you would any application installation.

After a list of applications is compiled, an analysis can be performed. ACT identifies the logged applications and checks them against Microsoft, vendor, and community databases. Checking against an ACT community database allows different organizations to rate the compatibility of applications and give other organizations useful compatibility information that is vendor and OS independent.

Lastly, ACT can create fixes for applications. Many of the fixes are similar to setting compatibility modes except they only implement the exact needed legacy emulation. This eliminates the situation in which a compatibility mode fixes the original problem but causes another to do unnecessary legacy emulation. Other fixes are permissions fixes that can be overcome in various ways. One permission fix ACT can apply is to redirect sections of the registry that are not writable by a standard user to sections, which are writable by the user. Vista has a dynamic ability to virtualize changes the user does not have permission to make. The changes are stored in a virtual overlay that is maintained in the user profile and is transparent to the user.

The Application Compatibility Toolkit is a very useful and powerful tool when used correctly. Although it's useful during a migration, it can continue to be used for service pack updates and Internet Explorer upgrades. When planning your deployment, be sure to consider ACT as a part of your data collection stage as it can help reduce problems and downtimes after deployment.

Installing and configuring the Application Compatibility Toolkit

The steps are provided as a quick start to installing, deploying, and reviewing results using the Application Compatibility Toolkit. If you do not have a good inventory of what software is in your environment, doing this can prove a valuable tool even outside mitigating compatibility issues. Follow these steps:

1. **Download the Application Compatibility Toolkit from the Microsoft Web site at** `http://go.microsoft.com/fwlink/?LinkID=82101`

   It is recommended that a server be used to host ACT as the limited number of inbound network connections on a workstation (ten) may be insufficient if many computers are attempting to report results simultaneously.

2. **Run through the simple installation wizard and launch the Application Compatibility Manager tool using the generated shortcut.** A configuration wizard appears on first launch with a Welcome screen that describes its purpose as to help you establish a database, create a share where client systems will place generated log files, and to establish a log processing service account to process collected logs.

3. **Click Next to begin.**

4. **Choose to configure the local computer to process logs and view reports (Enterprise Configuration) or to simply view and manage reports from an existing database (View and manage reports only). In this case, choose Enterprise Configuration and click Next.**

5. **Configure the ACT database. If you do not have a SQL server available, SQL Server Express (**`www.microsoft.com/sql/editions/express/default.mspx`**) may be used. Choose a SQL server from the list and click the Connect button.**

6. **After a successful connection has been established, you may enter the name of the ACT database that is to be created (for example ACTdata) and click the Create button.** Once the database has been successfully created, the Next button will be enabled.

7. **Click the Next button to continue.**

8. **Specify a path to an existing folder be used to hold the data generated by clients reporting inventory information.** You can use the Browse button to locate or create a folder, but it must exist before proceeding beyond this screen.

9. **Specify a local path and then enter the name that you would like to share.** By default, a share name will be generated based on the physical path specified in the Path field.

10. **Edit as desired (for example, ACTlogs) and click Next to continue.** The configuration wizard completes its tasks by requesting credentials to be used to run as a service which will process the log files. The account requires read/write access to the database and the log file share.

11. **You may specify a specific service account or accept the default Local System option. Click Next to continue.** A Congratulations screen appears to advise you of the successful completion of ACT configuration.

12. **Leave the Automatically check for updates when launching check box selected and click the Finish button to close the ACT configuration wizard.**

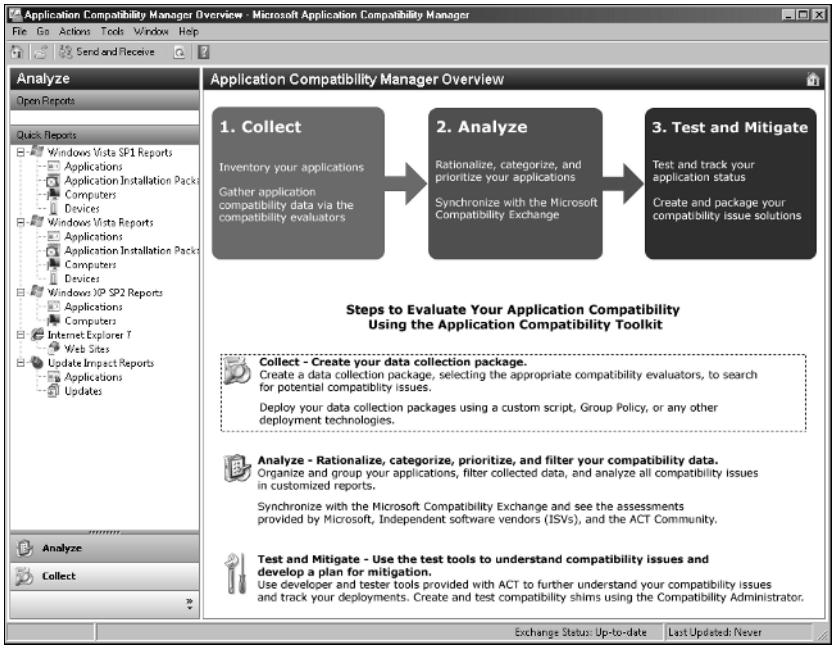
Creating a data collection package for ACT

Once installed, it is necessary to create a collection package which is an MSI setup that may be run on clients to have them report data to the Application Compatibility Manager for processing and reporting. The following steps cover the creation of the collection package.

The home screen for the Application Compatibility Manager (shown in Figure 1.8) presents a simple menu for completing each of the steps necessary to utilize ACT. This screen may also be accessed by clicking the overview toolbar button at the top left that is represented by an icon image of a house.

FIGURE 1.8

Initial launch of the Application Compatibility Manager



1. **Click the top item titled Collect – Create your data collection package to begin**. The new package screen appears which provides the following fields and options:
   - **Package Name**: Enter a friendly **name** for the package (for example ACTpackage).
   - **Evaluate compatibility when:** Three options are presented including Deploying a new OS or Service Pack (default), updating IE, or applying updates. Choose Deploying a new Operating System or Service Pack.

- **When to monitor application usage:** You can choose the starting time (default is "As soon as possible after installation"), the duration (default is three days), and frequency (the default period is eight hours).
- **Where to put collected data:** The default output location specified should already be that of the share specified during the previous configuration steps, but modify if necessary.

2. **When complete, click the save icon on the toolbar at the top left of the window.** A file location and filename prompt will allow you to choose the location and filename of the MSI setup which represents the data collection package for ACT.

Execute the collection package on client systems

There are a number of ways the collection package may be deployed from manual execution to deployment via an in-place systems management solution. As an MSI setup, Group Policy presents an attractive method of deployment for many organizations. The following steps take you through deploying the ACT package via Group Policy.

1. **Begin by running the Group Policy Management console (from the Administrative Tools program group on Windows Server).** The GPO object you choose will dictate what clients will install the package and report to ACT.
2. **Choose an existing GPO and choose Edit, or select a desired container, such as an OU and choose Create and Link a GPO Here.**
3. **If you create a new GPO, provide a descriptive Name such as ACT Client, click OK, right-click the newly created GPO, and choose Edit.**
4. **Choose ACT Client ➪ Computer Configuration > Software Settings, right-click Software installation, and then choose New ➪ Package**. A browse dialog will appear for you to specify the location of the MSI created in the previous steps.
5. **Select the MSI package and click OK.**
6. **In the resulting window, accept the default option of Assigned and click OK to continue.**

NOTE **The installation will take place without prompting for input, even if executed manually by double-clicking or running the installation from the command line.**

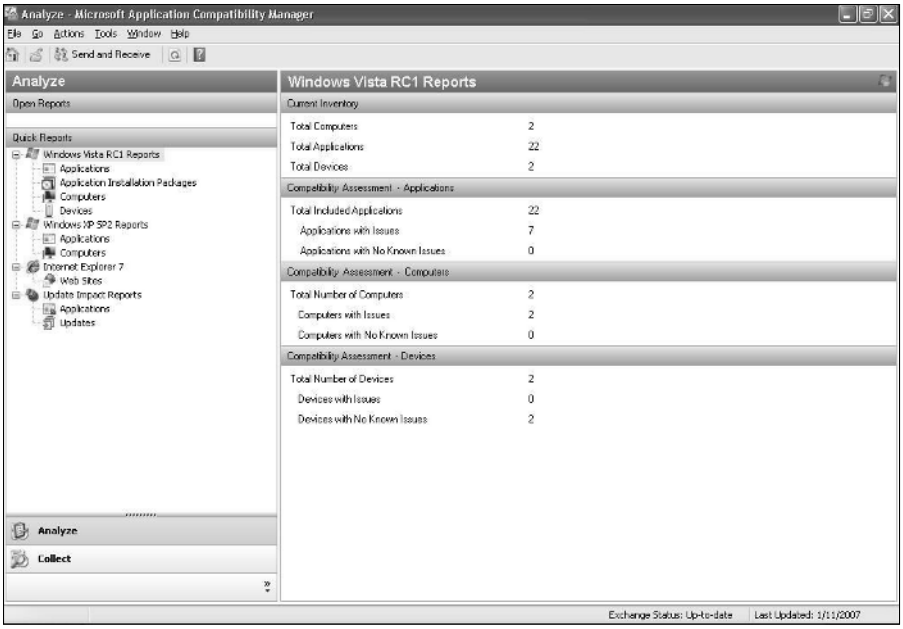
Reviewing ACT reports

After data has been collected, several reports can be viewed from the Application Compatibility Manager interface (see Figure 1.9).

Several reports are provided, and you can identify and even address compatibility issues from within this console. Click the Send and Receive toolbar button to begin the process of obtaining compatibility information on the applications detected in your environment. You can create remediation packages and even share application compatibility ratings with the community. For more details on ACT, visit `http://technet.microsoft.com/en-us/windowsvista/aa905102.aspx`.

FIGURE 1.9

Reviewing results collected by the deployed ACT collection packages



Documenting the Deployment Plan

When documenting your deployment plan, consider the following questions:

- What Edition(s) of Windows Vista should be used? Justify each.
- How many images will be supported? Justify each.
- Which applications will be included in the deployment images?
- Which applications will not be included and how will they be distributed?
- What type of distribution media will be used and why?
- Which hardware will be reused? Upgraded? Replaced?
- What critical applications are not Vista compatible by default and how can they be corrected?

Summary

Microsoft has significantly changed the installation process. Understanding the changes and taking advantage of them can greatly improve your organizational deployment of Windows Vista. The most drastic change implemented is the introduction of the WIM file images. However, WIM files only handle the problem of image storage and modification. Planning should be a very important part of your deployment process. Be sure to consider hardware requirements and upgrades as well as application compatibility. When proper planning has been done, the technologies in the rest of this book can used to simplify deployment and management as a well as increase the stability of your workstations.