|  |
| --- |
|  |

By   
Published: June 2007

Abstract

Understanding the inventory of deployment tools for Windows Vista™ is the first step toward successfully deploying the operating system. This white paper describes the tools built into Windows Vista and the Microsoft® Windows® Automated Installation Kit (Windows AIK). It also describes other critical deployment technologies, including Microsoft Windows Deployment Services (Windows DS) and Microsoft System Center Configuration Manager and how they fit into the overall Business Desktop Deployment 2007 solution.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2007 Microsoft Corporation. All rights reserved. Microsoft, Internet Explorer, MS-DOS, Win32, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners. Microsoft Corporation • One Microsoft Way • Redmond, WA 98052-6399 • USA

Contents

[Overview 1](#_Toc170182844)

[Windows Imaging 3](#_Toc170182845)

[Windows Vista 4](#_Toc170182846)

[Application Compatibility Toolkit 7](#_Toc170182847)

[User State Migration Tool 9](#_Toc170182848)

[Volume Activation 10](#_Toc170182849)

[Windows Vista Hardware Assessment 11](#_Toc170182850)

[Windows Automated Installation Kit 12](#_Toc170182851)

[Business Desktop Deployment 15](#_Toc170182852)

[Windows Deployment Services 19](#_Toc170182853)

[Systems Management Server 2003 and System Center Configuration Manager 2007 20](#_Toc170182854)

[Summary 22](#_Toc170182855)

[Glossary 23](#_Toc170182856)

# Overview

Understanding the new deployment tools and how they interconnect is the first step in beginning a Windows Vista™ deployment project. Microsoft has invested in providing a breadth of tools as well as the guidance to enable repeatable, scalable, and cost-effective deployments.

Figure 1 illustrates the Windows Vista deployment platform based on the Microsoft® Solution Accelerator for [Business Desktop Deployment (BDD) 2007](http://www.microsoft.com/desktopdeployment).



Figure 1. BDD 2007 deployment platform

At the lowest tier are Windows Imaging Format (WIM) files, which are highly compressed, file-based operating system images. Microsoft ships Windows Vista in Windows image files.

At the second tier are answer files. Earlier versions of the Microsoft Windows® operating system had numerous answer files, including Unattend.txt and Sysprep.inf, to drive the deployment process. Windows Vista uses a single Extensible Markup Language (XML)–based answer file, Unattend.xml, to drive all its configuration passes. This improvement makes configuration more consistent and simplifies engineering.

At the third tier are the various deployment tools for Windows Vista. The Windows Vista distribution media include some of these tools, such as Windows Setup, the System Preparation Tool (Sysprep), and other command-line tools. The Microsoft Windows Automated Installation Kit (Windows AIK) includes the larger tools, such as Microsoft Windows System Image Manager (Windows SIM), Microsoft Windows Preinstallation Environment (Windows PE) 2.0, and ImageX. These basic tools are necessary for creating, customizing, and deploying Windows Vista images. They are stand-alone tools that don’t provide a deployment framework or add business intelligence and best practice to the process, however.

The fourth tier, BDD 2007, provides the right framework, business intelligence, and best practices. BDD 2007 is a process and technology framework that uses all the tools in the third tier, helping you save hundreds of hours of planning, developing, testing, and deployment. BDD 2007 is based on best practices developed by Microsoft, its customers, and its partners. It includes time-proven management and technology guidance as well as thousands of lines of thoroughly tested script code that you can use as is or customize to suit your organization’s requirements.

Using BDD 2007, you can perform both Lite Touch Installation (LTI) and Zero Touch Installation (ZTI) deployments. LTI requires very little infrastructure and is suitable for most small and medium businesses. Microsoft Windows Deployment Services (Windows DS) helps make deployment more efficient by starting computers from the network. ZTI deployments take advantage of the Microsoft System Center Configuration Manager 2007 (or Systems Management Server 2003) infrastructure and are suitable for organizations that already have this enterprise-class infrastructure in place.

The following sections provide more information about the components shown in Figure 1.

# Windows Imaging

Windows Vista will be distributed in .wim files, the new image-file format. This format has the following advantages:

* Windows imaging files are a file-based image format that lets you store multiple images in one file. You can perform partial volume captures by excluding files (such as paging files) that you don’t want to deploy with the image.
* This format reduces file sizes significantly by using a compressed file format and single-instance storage techniques (the image file contains one physical copy of a file for each instance of it in the image file, which significantly reduces the size of image files that contain multiple images).
* You can service the image contained in the .wim file, including adding and deleting packages, software updates, and device drivers, without recreating a new image by applying it, customizing it again, and recapturing it. For more information, see the section “Other Deployment Tools” later in this white paper.
* You can mount .wim files as folders, making it easier to update files in images they contain.
* Windows Imaging files enable the nondestructive application of an image to the destination computer’s hard disk. You can also apply an image to different-sized destination disks since .wim files don’t require the destination hard disk to be the same size or larger than the source hard disk.
* Windows Imaging files can span media, enabling you to use CD-ROMs to distribute large .wim files.
* Windows PE .wim files are bootable. For example, you can start Windows PE from a .wim file. In fact, Windows Vista Setup and Windows DS start Windows PE from the Boot.wim file, which you can customize by adding items such as device drivers and scripts.

The WIMGAPI is the Windows Imaging application programming interface (API) that developers can use to manage .wim files. The API exposes all imaging functionality. In fact, ImageX is a command-line interface for WIMGAPI. For more information about ImageX, see the section “ImageX” later in this white paper.

The possibilities are endless. For example, a developer can use the API’s functions to create a new .wim file and capture an image to it. The developer’s code can mount the .wim file to a folder, update its contents, and unmount the .wim file. Finally, a developer’s code can prepare a computer’s hard disk by partitioning and formatting it, then applying the .wim file to the computer. The API even provides callback messages that enable the developer’s code to display progress and error messages to the user.

WIMGAPI enables independent software vendors (ISVs) to develop third-party image deployment and servicing products to fill almost any need. It also enables your company’s information technology (IT) developers to more easily create custom imaging solutions for your organization.

# Windows Vista

Many of the deployment tools for Windows Vista are included in the operating system. The two primary tools are Windows Setup, which drives the installation process, and Sysprep, which prepares master installations for imaging. The following sections provide an overview of these tools.

## Windows Setup

Windows Setup (Setup.exe) is the program that installs Windows Vista. It uses image-based setup (IBS) to provide a single, unified process with which all customers can install Windows. IBS performs clean installations and upgrades of earlier versions of the Windows operating system to Windows Vista. With Windows Setup and IBS, you can easily and cost-effectively deploy Windows Vista in your organization.

Windows Setup includes several new features that facilitate faster and more consistent installations:

* **Improved image management.** Windows Vista images are stored in a single .wim file. A .wim file can store multiple instances of the operating system in a single, highly compressed file. The installation file, Install.wim, is located in the Sources folder on the Windows Vista media. During installation, Windows Setup copies the contents of the image to the destination computer’s hard disk.
* **Streamlined installation.** Windows Setup is optimized to enable the deployment scenarios that most organizations use. Installation takes less time and provides a more consistent configuration and deployment process, resulting in lower deployment costs.
* **Faster installations.** Because Windows Setup is now image-based, installing and upgrading Windows Vista is faster and easier. You can perform clean installations of Windows Vista by deploying the Windows image to destination computers; you perform upgrades by installing a new image onto an existing installation of Windows. Windows Setup protects the earlier Windows settings during the installation.
* **More robust answer files.** With earlier versions of the Windows operating system, Windows Setup was automated by multiple answer files, including Unattend.txt, Sysprep.inf, and Winbom.ini. These answer files enabled automation during a particular phase of installation—one answer file for each phase. Windows Vista uses a single answer file, Unattend.xml, for all configuration passes, which leads to consistency across different passes. Windows Vista supports more settings in this XML-based answer file than earlier versions of the Windows operating system, making it easier to deploy the right configuration using nothing more than an answer file. You use the Windows SIM to edit Windows Vista answer files (see “Windows SIM”).

## System Preparation Tool

You use Sysprep to prepare a master installation for imaging and deployment. BDD 2007 automatically runs Sysprep to prepare the installation prior to capturing a disk image. Sysprep performs the following tasks:

* **Removes computer-specific data from Windows Vista.** Sysprep can remove all computer-specific information from an installed Windows Vista image, including the computer security identifier (SID). You can then capture and install the Windows installation throughout your organization.
* **Configures Windows Vista to boot to Audit Mode.** You can use Audit Mode to install non-Microsoft applications and device drivers as well as to test the functionality of the computer.
* **Configures Windows Vista to boot to Windows Welcome.** Sysprep configures a Windows Vista installation to boot to Windows Welcome the next time the computer starts. Generally, you configure a system to boot to Windows Welcome as a final step before delivering the computer to the user.
* **Resets Windows Product Activation.** Sysprep can reset Windows Product Activation as many as three times.

Sysprep.exe resides in the %SYSTEMROOT%\system32\sysprep directory on all Windows Vista installations. You don’t have to install Sysprep separately, as in earlier versions of Windows, because it’s a native part of the installation. You must always run Sysprep from the %SYSTEMROOT%\system32\sysprep directory on the version of Windows Vista with which it was installed.

## Other Deployment Tools

Windows Vista includes many of the command-line deployment tools natively. The following list describes the most commonly used tools:

* **Bcdedit.exe.** A command-line tool for managing Boot Configuration Data (BCD) stores. BCD files provide a store that describes boot applications and boot application settings. The objects and elements in the store replace Boot.ini. Bcdedit.exe has many uses, including creating new stores, modifying existing stores, adding boot menu options, and so on. Bcdedit.exe serves the same purpose as Bootcfg.exe on earlier versions of the Windows operating system, but it provides more options and is easier to script.
* **Bootsect.exe.** Updates the master boot code on hard disk partitions. Use this tool to switch between BOOTMGR and NTLDR. You also use this tool to restore the boot sector on a computer. Bootsect.exe replaces FixFAT and FixNTFS.
* **Diskpart.exe.** A text-mode command interpreter that enables you to manage disk-related objects (disks, partitions, or volumes) by using scripts or direct input at a command prompt.
* **Expand.exe.** Expands one or more compressed update files. Expand.exe supports opening updates for Windows Vista as well as earlier versions of Windows. By using Expand, you can open and examine updates for Windows Vista on a Microsoft Windows XP or Windows Server® 2003 operating system. The Windows AIK also includes this tool so you can use it on versions of Windows earlier than Windows Vista.
* **Lpksetup.exe.** Performs unattended language-pack operations. Lpksetup.exe can install and uninstall language packs silently, for example. This tool only runs online—you can’t use it to add language packs to an offline Windows Vista image.
* **Pkgmgr.exe.** A command-line tool that you can use offline to install, remove, or update Windows Vista packages. You can add packages, provided as .cab files, to an offline Windows Vista image. Pkgmgr.exe (Package Manager) can enable or disable features, either offline or on a running installation. You can use this tool to do the following with offline Windows Vista images:
* Install or uninstall updates that Microsoft provides.
* Install language packs.
* Add out-of-box drivers to the driver store.
* Enable or disable Windows features.
* Accept an answer file as input (offlineServicing settings only).
* Add packages to an offline Windows image.
* Install or uninstall multiple packages with one command string.
* **Powercfg.exe.** Controls power settings and configures computers to default to Hibernate or Standby modes. You can use this tool to fully script power-management configurations during deployment.

# Application Compatibility Toolkit

The Application Compatibility Toolkit (ACT) 5.0 provides tools for creating an application inventory, identifying compatibility issues, and mitigating those issues.

The software requirements for the ACT are straightforward. It requires at least Windows XP Professional with Service Pack 1 (SP1) or a later version of the Windows operating system. You must have a database in which to store application data. You can use Microsoft SQL Server™ 2005, SQL Server 2005 Express, or SQL Server 2000. You must also install the Microsoft .NET Framework 1.1 on the server computer.

After installing and configuring ACT, use the following four-step process to create an application inventory for your environment:

* **Inventory.** During the inventory process, you deploy the ACT agent to computers running at least Microsoft Windows 2000 with SP4 or a later version of the Windows operating system. You can deploy the agent by using a logon script or your existing systems management infrastructure. The agent gathers an inventory from each computer by looking at different installation areas. ACT includes Windows Vista–specific compatibility evaluators to help you identify issues in your application inventory. For example, the Windows Vista User Account Control (UAC) Compatibility Evaluator identifies compatibility issues resulting from permission restrictions that UAC enforces.
* **Analyze.** After collecting an inventory, the next step is to analyze it. During this analysis, you gather metadata about each application in the inventory that will help you develop a strategy around how you will organize the applications. ACT allows you to create custom compatibility reports that meet your own requirements. You can also assign custom categories for your applications that help you add more meaning to your requirements. For example, you can create categories for geographies, departments, and vendors.
* **Rationalize.** After analyzing the application inventory, you rationalize it by refining the list. Your organization might have hundreds if not thousands of applications. However, you can’t worry about every application in the organization. In this phase, you reduce the list of applications you’re examining to focus your efforts on the correct set of applications. ACT includes new filtering features that help you refine your application inventory. For example, you can eliminate irrelevant applications, applications that have specific issues, or applications with no compatibility information. You can also find and share compatibility information with industry peers through the Online Compatibility Exchange.
* **Prioritize.** After narrowing down the list of applications by rationalizing it, you can further refine the list by prioritizing each application. The goal is to focus on delivering a smooth and almost transparent operating system deployment. ACT helps you prioritize your applications and identify those applications that are mission-critical. ACT also provides basic workflow management by helping you track the deployment process with status levels such as Not Tested, In Testing, In Mitigation, and Ready to Deploy.

The latest version of ACT is much more than just a tool for testing application compatibility. It’s now a life cycle–management tool for applications, enabling you to gather an inventory into a complete picture of the applications in your organization—an application portfolio. Even after you deploy the operating system, ACT will continue to be a valuable part of your IT operations.

For more information about ACT, see **Microsoft Application Compatibility Toolkit 5.0** at <http://technet.microsoft.com/en-us/windowsvista/aa905102.aspx>. BDD 2007 provides guidance for using ACT within the scope of a larger deployment project and the BDD solution enables you to download and install this toolkit.

# User State Migration Tool

Operating system deployment always involves user state migration—the process of migrating users’ documents and settings from one operating system to another. Even when you don’t migrate user state during deployment, users will spend countless hours trying to restore their preferences (desktop backgrounds, screensavers, and themes). Because this manual process reduces user productivity and usually increases support calls (because users often forget how to configure settings), organizations often choose to migrate some portion of user state to new operating systems as they’re deployed.

User satisfaction is another reason to elevate the importance of user state migration in your project. Users are simply more satisfied and feel less overwhelmed when they sit down in front of a new operating system and they don’t have to recover their preferences. The fact is that unsatisfied users can lead to poor post-implementation reviews and have negative consequences for future deployment projects. For example, user dissatisfaction with previous projects can stall a deployment project that you know will benefit the company in the long term.

Use Microsoft Windows User State Migration Tool (USMT) to migrate user state in high-volume deployment projects. It can execute complex, repeatable migrations of user state data between operating systems. You can script USMT, or you can execute it directly in a Command Prompt window. BDD 2007 automatically runs USMT during LTI and ZTI deployments. Of course, BDD 2007 provides options for customizing how it runs USMT.

In addition to document and settings migration, USMT can migrate application preferences for Microsoft Office applications between versions of Microsoft Office. For example, USMT can migrate Microsoft Office XP or later settings to newer versions of Microsoft Office, including the 2007 Microsoft Office system.

# Volume Activation

Microsoft Volume Activation 2.0 is technology for activating Windows operating systems and, in the future, other Microsoft programs such as Microsoft Office, ensuring that both Microsoft and the customer are protected from casual copying. Through the creation of limited-use product keys (called Multiple Activation Keys, or MAKs) or by requiring systems to periodically renew their activation using a Key Management Service (KMS) infrastructure, Microsoft gives organizations a solution that may protect their license investments. This solution has several benefits, in addition to helping solve the problems of intellectual property theft and customers’ loss of product support:

* **Flexible, easy activation options.** The new KMS infrastructure is simple to operate, requiring little time for proper configuration and activation. Systems imaged for this environment automatically maintain their activation with no additional effort on the part of IT administrators. MAKs are easy to distribute, and the entire process is invisible to users no matter which option is used.
* **Better software asset management.** Although Volume Activation is in no way tied to the billing process, administrators will be able to generate reports on software activation by using a provided reporting tool, a Microsoft Operations Manager (MOM) pack, or a non-Microsoft license auditing tool. By knowing the number of activated products in their environment, administrators can monitor volume license usage and know when to budget for additional licenses. Microsoft’s online license management portals allow administrators to request additional keys to activate the computers purchased to replace stolen systems or to reactivate systems that have gone out of tolerance or have been re-imaged.
* **Better Volume License Key management tools.** Volume License Keys are now encrypted and kept in a trusted store on the client, and in general they are easy for IT professionals to manage and protect, especially in the case of KMS. Using BDD 2007 tools, administrators can prepare reference systems for image capture. Administrators can activate Windows by using scripts after systems are imaged, providing automation of activation and helping to reduce administrative effort. The Microsoft Volume Activation Management Tool (VAMT) is part of Volume Activation and allows organizations to manage the activation of their Windows Vista computers using MAKs. These new capabilities allow much tighter control of activation keys. Organizations can now ensure that their activation keys are used only to activate their systems without extensive custody control processes.
* **Better protection for data on lost and stolen systems.** Systems activated with KMS must periodically renew their activation against the customer’s hosted KMS to remain in operation. Systems taken from environments protected by this system eventually revert to Reduced Functionality Mode (RFM), limiting their usefulness and making them less attractive to theft.

For more information about volume activation, see **Volume Activation Guide** at <http://www.microsoft.com/technet/desktopdeployment/bdd/2007/VolumeAct.mspx>.

# Windows Vista Hardware Assessment

Windows Vista Hardware Assessment (WVHA) is a centralized and agentless tool that assesses the readiness of computers for Windows Vista. It quickly inventories computers, identifies the supported Windows Vista experience, and recommends specific hardware upgrades. WVHA runs on a single networked computer to assess all computers on the network.

WVHA provides detailed reports that describe the readiness of existing computers for Windows Vista. It describes each computer’s readiness using the following three levels:

* Computers that are incompatible with Windows Vista
* Computers that are ready for the Windows Vista Capable experience
* Computers that are ready for the Windows Vista Premium Ready experience

In addition to reporting the readiness of existing computers, WVHA recommends specifications that you must address to get a computer ready for Windows Vista installation to improve the user’s experience. For example, WVHA might recommend that you upgrade a computer to 512 megabytes (MB) of memory to get the Windows Vista Capable experience. It might recommend that you upgrade the graphics card to get a Windows Vista Premium experience.

WVHA provides you with clear guidance on the computer assets you have and their capability to run Windows Vista. It also provides recommendations for computers that are not capable of running Windows Vista. It’s an unobtrusive, centralized method for capturing this information, and it complements tools like ACT 5.0 and BDD 2007. For more information about WVHA, see <http://www.microsoft.com/technet/wvha>.

# Windows Automated Installation Kit

The Windows AIK includes Windows PE, Windows System Image Manager, and other Windows Vista deployment tools. Using these tools together, you can customize and deploy Windows Vista. BDD 2007 fully leverages all these tools, however, and it applies best practices and business intelligence to them. In fact, you can download and automatically install Windows AIK by using BDD 2007. Abstracting these tools by using BDD 2007 is a better way to deploy Windows Vista than using these tools directly. For more information about Windows AIK, see <http://technet.microsoft.com/en-us/windowsvista/aa905061.aspx>.

## Windows PE

Prior to Windows PE, organizations often had to use MS-DOS® boot floppy disks to start destination computers, and then start Windows Setup from a network share or other distribution media. MS-DOS boot floppy disks had numerous limitations, however, including no support for the NTFS file system, no native networking support, and a requirement to locate 16-bit device drivers that worked in MS-DOS.

Windows PE 2.0 provides a minimal Win32® operating system with limited services—built on the Windows Vista kernel—that you use to prepare a computer for Windows Vista installation, copy disk images to and from a network file server, and start Windows Setup. Windows PE 2.0 is designed solely as a stand-alone preinstallation environment and as an integral component of other setup and recovery technologies, such as Windows Setup, Windows DS, Configuration Manager 2007, SMS Operating System Deployment (OSD) Feature Pack, and BDD 2007. Unlike earlier versions of Windows PE, which were only available as a Software Assurance (SA) benefit, Windows PE 2.0 is now publicly available in the Windows AIK.

Windows PE provides the following features and capabilities:

* Native support for NTFS 5.x file system, including dynamic volume creation and management
* Native support for TCP/IP networking and file sharing (client only)
* Native support for 32-bit (or 64-bit) Windows device drivers
* Native support for a subset of the Win32 API; optional support for Windows Management Instrumentation (WMI) and Windows Script Host (WSH)
* Can be started from multiple media, including CD, DVD, USB Flash Drive (UFD), and Windows DS

Windows PE runs every time you install Windows Vista, whether you install the operating system by starting the computer with the Windows Vista DVD or deploy Windows Vista from Windows DS. The graphical tools that collect configuration information during the setup phase run within Windows PE. In addition, you can customize and extend Windows PE to meet specific deployment needs. For example, BDD 2007 customizes Windows PE for LTI deployments by adding device drivers, deployment scripts, and so on.

When using BDD 2007, you don’t have to build or customize Windows PE manually. BDD 2007 automatically generates fully customized Windows PE images that you can burn to DVDs or add to a Windows DS server. BDD 2007 allows you to refine these Windows PE images by providing several options, such as choosing the device drivers to include, choosing optional fonts to include, and changing the Windows PE background.

## Windows System Image Manager

You use Windows System Image Manager (SIM) to create and configure Windows Vista answer files. Windows Setup uses answer files you create by using Windows SIM to configure and customize the default Windows Vista installation for all configuration passes. For instance, you can customize Windows Internet Explorer®, configure Windows Firewall, and specify the hard disk drive configuration. You can use Windows SIM to customize Windows Vista in the following ways and more:

* Install non-Microsoft applications during installation.
* Customize Windows Vista by creating answer files (Unattend.xml).
* Apply language packs, service packs, and updates to an image during installation.
* Add device drivers to an image during installation.

BDD 2007 automatically generates answer files that fully automate the Windows Vista installation process. It’s possible that you won’t need to further customize the answer file when using BDD 2007. However, BDD 2007 provides easy access to Windows SIM when you create an operating system build, which allows you to further customize the answer file associated with the build.

## ImageX

ImageX is an easy-to-use command-line utility that you use to create and manipulate Windows Imaging files. With ImageX, you can capture images and apply them to prepared destination computers’ hard disk drives. You can mount .wim image files as folders and thereby edit images offline. ImageX addresses the challenges that organizations face when using sector-based imaging formats or the MS-DOS **XCopy** command to copy an installation of Windows onto new hardware. For example, sector-based imaging:

* Requires that the destination computer use the same Hardware Abstraction Layer (HAL) as the master computer.
* Requires that the destination computer boot from the same type of mass-storage controller as the master computer.
* Destroys the existing contents of the destination computer’s hard disk drive, complicating migration scenarios.
* Duplicates the hard disk drive exactly; therefore, the image can deploy only to partitions that are the same type and at least as large as the source partition on the master computer.
* Does not allow for direct modification of image file contents.

The limitations of sector-based imaging led Microsoft to develop a file-based imaging solution, ImageX, and the accompanying .wim image file format. You can use ImageX to create an image, modify the image without going through the extraction and re-creation process, and deploy the image to your environment—all using the same tool.

Because ImageX works at the file level, it provides numerous benefits. It provides more flexibility and control over your images. For example, you can mount an image onto a folder, and then add files to, copy files from, and delete files from the image using a file-management tool such as Windows Explorer. ImageX allows for quicker deployment of images and faster installations than with earlier versions of the Windows operating system. With the file-based image format, you can also deploy images nondestructively so that ImageX does not erase the destination computer’s hard disk drive.

ImageX supports highly compressed images. First, .wim files support single instancing: File data is stored separately from path information so if duplicate files exist across multiple paths, the file data is stored one time, saving space. Second, .wim files support two compression algorithms—fast and maximum—which give you control over the size of your images and the time required to capture and deploy them.

BDD 2007 uses ImageX to capture images of master installations. It also uses ImageX to optionally back up the destination computer before installing an operating system build to it.

## Other Tools

Windows AIK includes many other deployment tools, including the following:

* **Drvload.exe.** Adds out-of-box drivers to a booted Windows PE image. It takes one or more driver .inf files as inputs. To add a driver to an offline Windows PE image, use the Peimg.exe tool.
* **Intlcfg.exe.** Changes the language, locale, fonts, and input settings of a Windows Vista image. Typically, you run Intlcfg.exe after applying one or more language packs to a Windows Vista image.
* **Oscdimg.exe.** Creates an image (.iso) file of a customized 32-bit or 64-bit version of Windows PE. You can then burn that .iso file to a CD-ROM or DVD-ROM.
* **Peimg.exe.** Creates and modifies Windows PE 2.0 images offline. You use Peimg.exe to import and install optional packages to Windows PE. You also use it to add device drivers, choose a language, and optimize a Windows PE image.
* **WIMGAPI.** The Windows Imaging API that developers can use to manage .wim files. The API exposes all Windows imaging functionality. In fact, ImageX is a command-line interface for WIMGAPI. Windows AIK includes file system driver for WIMGAPI that you can use with versions of the Windows operating system earlier than Windows Vista. Windows Vista already includes WIMGAPI.
* **Wpeinit.exe.** Initializes Windows PE each time it boots. When Windows PE starts, Winpeshl.exe executes Startnet.cmd, which launches Wpeinit.exe. Wpeinit.exe installs Plug and Play devices, processes Unattend.xml settings, and loads network resources.
* **Wpeutil.exe.** Enables you to run various commands in a Windows PE session. For example, you can shut down or restart Windows PE, enable or disable firewalls, and initialize a network.

# Business Desktop Deployment

BDD 2007 is a holistic approach to desktop deployment, bringing together the people, processes, and technology required to perform highly successful, repeatable, and consistent deployment projects. Because of its strong focus on methodology and best practices, BDD 2007 is much more valuable than the sum of its parts. Not only does BDD 2007 have the benefit of decreasing the time required to develop a desktop-deployment project, but it also reduces errors and helps you create a higher-quality desktop-deployment project.

Microsoft has been working with industry experts, system integrators, and deployment/management software providers in building this guidance so that it captures best practices from throughout the industry. The result is the significantly improved Business Desktop Deployment (BDD 2007) methodology for desktop deployment.

Microsoft has two implementations of BDD methodology: Lite Touch (LTI) and Zero Touch (ZTI). The Lite Touch version of BDD is most useful for companies that might not have a management infrastructure but want to take advantage of the solution to do a full, end-to-end deployment. Microsoft's recommended and high-end BDD solution is the Zero Touch implementation, using the enterprise-class infrastructure that SMS 2003 provides.

BDD 2007 includes an information center that helps you get easily started with the solution. The information center provides easy access to all of the solution’s guidance. It also allows you to download and automatically install all of the components that BDD 2007 requires, including Windows AIK, ACT5.0, WVHA, Volume Activation tools, USMT, Office migration tools, and required updates.

Microsoft recommends that you use BDD 2007 to deploy Windows Vista instead of using the basic deployment tools directly. All the deployment tools in Windows Vista and Windows AIK are huge improvements over the deployment tools for earlier versions of the Windows operating system. However, they are simply tools without a framework and without any business logic. They have no *glue* to bind them into an end-to-end process. BDD 2007 provides this glue in the form of a complete technology framework. Internally, BDD 2007 is an extremely sophisticated solution. It provides solutions for the problems facing most customers during deployment, including preinstallation phases (disk partitioning, formatting, and so on), installation (disk imaging), and post-installation phases (user state migration, application installation, customization, and so on). Even though BDD 2007 is complex internally, the solution makes building, customizing, and deploying Windows Vista images easy by masking most of the details.

Figure 2 describes the typical process for using BDD 2007 to deploy Windows Vista. The process is the same whether you’re capturing an image in the lab or deploying images in a production environment. In addition, BDD 2007 provides a user interface (UI) to configure all its processes. Behind the scenes, thousands of lines of code work to implement your choices during deployment.



Figure 2. BDD 2007 development and deployment process

The following list describes each part of the BDD 2007 deployment process:

* **Information Center**. After installing BDD 2007 on a build server in a lab environment, you can use the information center to read the recommended guidance and methodology. You can also use the components page to get and install all the latest tools, including all the free tools mentioned in this white paper.
* **Distribution share.** You first use the BDD 2007 Deployment Workbench to stock the distribution share with source files that you’ll use to build images for your environment. Source files include Windows Vista images, applications, device drivers, and packages. Deployment Workbench provides a UI for adding all source files to the distribution share. The UI also provides intelligence, such as error checking and building a device driver database for device driver injection during deployment.
* **Build.** After the distribution share is fully stocked, you use Deployment Workbench to create a build. A build associates source files from the distribution share with a configuration that customizes the operating system installation. The configuration includes a setup answer file (Unattend.xml) and a task sequence. A *task sequence* is a list of steps to take during installation. The task sequence specifies when to take each step and when to skip it. The task sequence supports restarts during installation, and data collected during the task sequencer persists between restarts. The task sequence represents one of the primary customization points for BDD 2007.
* **Deployment point.** After creating a build, you create deployment points for installing it. A *deployment point* is a subset of the BDD 2007 distribution share in much the same way that configuration sets are subsets of Windows SIM distribution shares.  
    
  A deployment point specifies which source files and builds from the distribution share to distribute and how to distribute them. Deployment points also provide a way for users to connect and begin installation. For example, a network deployment point copies a subset of the distribution share to another network location and generates a Windows PE boot image that automatically connects to the distribution share and begins installation. A Microsoft Systems Management Server (SMS) OSD Feature Pack deployment point creates a subset of the distribution share that you can deploy by using a Configuration Manager 2007 OSD Feature Pack deployment program. A special deployment point, called a *Lab deployment point,* always points to the full distribution share on the build server.  
    
  Deployment points are customizable. The primary customization points for deployment points are *rules,* which are stored in a file called CustomSettings.ini. This file contains rules for installing Windows Vista on destination computers and configuring it. You can configure the BDD 2007 Windows Deployment Wizard UI, for example, by hiding certain pages and automating others. You can also target specific Windows Vista settings at different groups of computers based on Media Access Control (MAC) address, location, and so on.

It is worth noting that this is an infrastructure that is separate from a traditional SMS 2003 or System Center Configuration Manager network hierarchy. In the ZTI approach, SMS 2003 OSD or SCCM OSD are used that take advantage of a deployed network infrastructure, including the benefits of SMS site replication, centralized management, and granular control of build distribution. In SCCM, network efficiency from Binary Delta replication ensures that your network only replicates changes that are made to larger OS packages.

* **Destination computer.** With a fully stocked distribution share, a defined build, and a defined deployment point, you can use BDD 2007 to deploy Windows Vista to destination computers. You can use an LTI or a ZTI deployment to deploy Windows Vista. To use LTI, you start the destination computer using the deployment point’s Windows PE boot image. You can put the boot image on removable media (DVD, UFD, and so on) or add it to a Windows DS server. Either way, you start the destination computer using the Windows PE boot image that the deployment point provides to begin the Windows Deployment Wizard. The wizard displays several screens to collect data from you (computer name, domain membership, applications to install, and so on), and then the wizard installs the operating system without any further interaction. You can also use ZTI deployment to deploy Windows Vista. To deploy Windows Vista using ZTI, you must utilize an SMS OSD Feature Pack deployment point.

This infrastructure is similar in SCCM, the primary difference being that for SCCM, OS Deployment is an out-of-the-box feature included in the product. There are also significant improvements over all of today’s OS deployment methodology within SCCM, including WinPE boot support, increased scenario support, PXE integration, Task Sequences, Driver catalogue services, and full integration to Windows Vista and Windows Server 2008 AIK.

* Figure 2 makes no reference to creating a master installation and capturing an image. In BDD 2007, creating and capturing an image is an LTI process. You can configure any deployment point to automatically capture an image of an installation and store the image in the distribution share. After you make this choice, the imaging process is fully automated. You don’t have to run Sysprep. You don’t have to run ImageX. The Windows Deployment Wizard automatically runs Sysprep, and then it runs ImageX to capture the image and store it in the distribution share. Then, you can simply add the image to the distribution share using Deployment Workbench.

See <http://www.microsoft.com/desktopdeployment/> for more information about BDD 2007.

# Windows Deployment Services

Windows Deployment Services (Windows DS) is the updated and redesigned version of Remote Installation Services (RIS) in Windows Server 2008. A Windows DS update is also available for Windows Server 2003 SP1. Windows AIK includes this update. Windows Server 2003 SP2 includes Windows DS as an operating system component. Windows DS helps organizations rapidly deploy Windows operating systems, particularly Windows Vista. Using Windows DS, you can deploy Windows operating systems over a network without using the media.

Windows DS delivers a better in-box deployment solution than RIS. It provides platform components that enable you to use custom solutions, including remote boot capabilities; a plug-in model for Pre-Boot Execution Environment (PXE) server extensibility; and a client-server communication protocol for diagnostics, logging, and image enumeration. Also, Windows DS uses the Windows Imaging file format and provides an improved management experience through Microsoft Management Console (MMC) and scriptable command-line tools.

For organizations that have already deployed RIS, Windows DS maintains parity with RIS by providing both coexistence and migration paths for RIS. Windows DS will continue to support RIS images in legacy or mixed mode. Windows DS also provides tools to migrate RIS RIPREP images to the new .wim image file format.

BDD 2007 generates Windows PE images that you use to connect to a distribution share and install Windows Vista. You can add these Windows PE images to Windows DS to make starting Windows PE quicker and easier than starting Windows PE from removable media.

See **Deploying Windows Vista** at <http://technet.microsoft.com/en-us/windowsvista/aa905061.aspx> for more information about Windows DS.

# Systems Management Server 2003 and System Center Configuration Manager 2007

Systems Management Server and the new Configuration Manager 2007 provide a comprehensive solution for change and configuration management for Microsoft platforms, enabling organizations to quickly and cost-effectively provide operating systems, software, and updates to users. The desktop deployment capability of these solutions is based on the BDD 2007 methodology. They give you a solution for planning, testing, deploying, analyzing, and optimizing your software using a familiar Windows interface.

### Systems Management Server 2003

Microsoft Systems Management Server (SMS) 2003—the update, deployment, and inventory management solution—combines with the deployment tools in Windows Vista and Office 2007 to efficiently and reliably deploy to your client computers, implementing the BDD end-to-end guidance for large-scale deployments.

You can deploy applications on your network in several ways: through CD-based installation, through script-based installation, with Microsoft Active Directory® Group Policy software deployment, or with SMS, which is the most flexible deployment method. SMS gives you control over the targeting of users or systems, the elevation of user rights for installation, and the ability to specify when an installation can take place. It also lets you determine user interaction levels, controlled restarts, and security accounts.

SMS helps you better plan your deployment by providing you with a detailed hardware and software inventory. You can obtain information about your hardware base, existing applications, version information, and current service pack and hotfix levels. Software distribution can then be specifically targeted to users and to computers based on their network and hardware configuration, Active Directory organization unit, and group membership and software installation status.

SMS 2003 can further simplify and streamline the deployment process for you with its Operating System Deployment (OSD) Feature Pack Update and with the Solution Accelerator for Business Desktop Deployment (BDD). The SMS 2003 OSD Feature Pack Update is a publicly available download that you can easily install on your SMS primary site server. It provides a customizable, centralized, and scalable way to create and deploy x32-bit and x64-bit client operating systems. The OSD Feature Pack simplifies Windows Vista deployment; it enables you to deploy Windows Vista in much the same way as you deploy other software applications. The SMS OSD Feature Pack Update uses an open architecture and is fully integrated with SMS. It provides an inventory based on planning and targeting, and it provides centralized tracking and status messages. With SMS, it manages the replication of images throughout distributed enterprises.

Out of the box, the OSD Feature Pack Update supports bare-metal and refresh scenarios. The OSD Feature Pack Update includes a version of Windows PE and includes features for capturing your master image. The OSD Feature Pack Update steps you through the process of capturing and managing the settings for individual deployments, including notifications, distribution settings, and network settings. The OSD Feature Pack Update uses the new Microsoft Windows Imaging (WIM) format.

The OSD Feature Pack Update integrates with the User Migration Tool (USMT) to preserve user profiles during the Windows Vista installation. The entire migration process is driven by the migration rules and logic that are built into the tool. You can specify all or none of the default files on the command line, depending on the particular migration. The OSD Feature Pack Update also reports on specific deployments to help troubleshoot and to demonstrate the success of the Windows Vista deployment.

**System Center Configuration Manager 2007**

System Center Configuration Manager 2007 (SCCM) provides a major upgrade for OS deployment functionality; it features significant enhancements from the SMS OSD Feature Pack Update and it brings in server deployment scenarios from Automated Deployment Services (ADS). There are many deployment scenarios possible with Configuration Manager: new machine, wipe-and-load, side-by-side, in-place migration, offline with removable media, and Pre-Boot Execution Environment (PXE) boot. Configuration Manager OS deployment provides enhancements for numerous scenarios—built-in integration with Windows Deployment Services for bare-metal deployment with PXE, and State Migration Points (SMP) as file shares to migrate user state and settings for side-by-side deployment. Configuration Manager provides device driver management, which provides generic OS images with separate installation of the device drivers needed by each client computer. It also adds enhanced security and extensibility.

System Center Configuration Manager is an enterprise-class management toolset and, as a result, all of the existing features of management are exposed to the Administrator throughout the deployment. Complete integration with software distribution, patch management, driver management, user state migration, the Vista deployment toolset (AIK), and the BDD methodology allows a full life-cycle approach to any scale deployment.

The System Center Configuration Manager operating system deployment (OSD) feature provides automated and hands-off deployment of Windows XP, Windows Vista, Windows Server 2003, and Windows Server 2008 operating systems to x86, x64, and IPF servers. It supports a variety of deployment scenarios, including wipe-and-load upgrade of an existing server, bare-metal provisioning from removable media or by using PXE via integration with WDS, and offline provisioning from removable media in branch offices with limited network bandwidth. OSD is based on the deployment technologies in Windows Vista and Windows Server 2008, including the WIM format, Windows DS, Windows PE, and sysprep. Thus, interchangeability of file images and tools is preserved regardless of which Microsoft deployment method is used.

When used in combination with other System Center Configuration Manager capabilities, such as Software Distribution, Software update services, Desired Configuration Management, Asset Intelligence, Internet-based Management, Branch Services, Network Access Protection and Device Management, the solution is robust, scalable, and centrally managed, answering the needs of even the largest organization.

For more information visit [www.microsoft.com/technet/sms](http://www.microsoft.com/technet/sms)

# Summary

BDD 2007 provides the framework, guidance, and business intelligence that brings together all the Windows Vista deployment tools. By using BDD 2007 and the tools that this white paper describes, you can reduce the time required to deploy Windows Vista. To learn how the Instituto Nacional de Estadística Geografía e Informática reduced deployment time by 80 percent, see the case study at <http://www.microsoft.com/casestudies/casestudy.aspx?casestudyid=200234>.

To preview the BDD 2007 guidance, see **Microsoft Solution Accelerator for Business Desktop Deployment 2007** at <http://www.microsoft.com/desktopdeployment>. You can download BDD 2007 from the same Web page. After installing BDD 2007, it can automatically download and install the other tools that this white paper describes.

# Glossary

The following terms are unique to Windows Vista deployment and BDD 2007. Understanding this terminology will help you better understand the deployment content in this book and the resources to which it refers.

* **Answer file.** A file that scripts the answers for a series of dialog boxes during installation. The answer file for Windows Setup is often called Unattend.xml. You can used Windows SIM to create and modify this answer file.
* **Build.** In BDD 2007, a build associates an operating system with a configuration containing an answer file (Unattend.xml) and a sequence of tasks to run on the destination computer during installation.
* **Catalog file.** A binary file that contains the state of all of the settings and packages in a Windows Vista image. When you use Windows SIM to create a catalog file, that file queries the Windows Vista image for a list of all settings in that image. Because the contents of a Windows Vista image can change over time, it is important that you re-create the catalog file whenever you update an image.
* **Component.** A part of the Windows Vista operating system that specifies the files, resources, and settings for a specific Windows Vista feature or part of a Windows Vista feature. Some components include unattended installation settings, which you can customize by using Windows SIM.
* **Configuration pass.** A phase of Windows Vista installation. Different parts of the Windows Vista operating system are installed and configured in different configuration passes. You can specify Windows Vista unattended installation settings to be applied in one or more configuration pass.
* **Configuration set.** A file and folder structure that contains files that control the preinstallation process and define customizations for the Windows Vista installation.
* **Deployment point.** In BDD 2007, a deployment point contains the collection of source files necessary to install builds and provides a mechanism for installing builds from the deployment point (Windows PE boot images).
* **Destination computer.** The computer on which you install Windows Vista during deployment. You can either run Windows Setup on the destination computer or copy a master installation onto a destination computer.
* **Distribution share.** A folder that contains the source files for Windows products that you install. It may also contain additional device drivers and application files. You can create this folder manually or by using Windows SIM. In BDD 2007, the distribution share contains operating system, device driver, application, and other source files that you configure by creating builds and distributing them through deployment points.
* **Image-based setup.** A setup process based on applying a disk image of an operating system to the computer.
* **Master computer.** A fully assembled computer containing a master installation of Windows Vista.
* **Master image.** A collection of files and folders (sometimes compressed into one file) captured from a master installation. This image contains the base operating system as well as additional configurations and files.
* **Master installation.** A Windows Vista installation on a master computer to be captured as a master image. You create the master installation using automation to ensure a consistent and repeatable configuration each time.
* **Package.** A group of files that Microsoft provides to modify Windows Vista features. Package types include service packs, security updates, language packs, and hotfixes.
* **Task sequence.** A sequence of tasks to run on a destination computer to install Windows Vista and applications, and then configure the destination computer. In BDD 2007, the task sequence is part of a build, and the component responsible for executing the task sequence is the Task Sequencer.
* **Task Sequencer.** The BDD 2007 component that runs the task sequence when installing a build.
* **Technician computer.** The computer on which you install BDD 2007 or Windows SIM. This computer is typically located in a lab environment, separate from the production network. In BDD 2007, this computer is usually called the *build server.*
* **Unattend.xml.** The generic name for the Windows Vista answer file. Unattend.xml replaces all the answer files in earlier versions of the Windows operating system, including Unattend.txt and Winbom.ini.
* **.wim.** A file name extension that identifies Windows image files that ImageX creates.
* **Windows image file.** A single compressed file containing a collection of files and folders that duplicate a Windows installation on a disk volume.
* **Windows Vista feature.** An optional feature of Windows Vista that you can enable or disable.