The Essentials Series: Automation Tools for Windows 7 Migration

Untangling Microsoft's Alphabet Soup of Migration Tools

sponsored by

KACE™
Systems Management. **Done.**

by Greg Shields

Copyright Statement

Untangling Microsoft's Alphabet Soup of Migration Tools

Most IT organizations quickly discover two hurdles in beginning an OS migration. The first is in finding the true business benefits of a new operating system (OS). Those benefits can't be expressed in terms of raw features; they must directly relate to measurable improvements in business processes. The second is in containing the costs associated with the migration itself. Keeping these costs low is critical as businesses tend to view migration activities as sunk costs.

Yet what many IT organizations don't realize is that these two seemingly-separate hurdles are in fact quite interrelated. First and foremost, businesses must see a measurable improvement in value if they're to make the jump. At the same time, the cost to transition mustn't negate any benefits gained by jumping in the first place. As a result, many IT organizations find themselves steering towards free tools to cheapen the cost of migration, only to later discover that the tools' limitations end up costing more.

The central problem is that many migration activities aren't well-planned from the start. OS migrations are uncommon occurrences in IT's life cycle. This means that internal organizations often don't have the background and experience to ensure the highest levels of success.

It is for these reasons you find so many solutions available that assist with OS migration. Even Microsoft itself recognizes the value in streamlining the migration process, developing and releasing a suite of no-cost tools that support each of the seven key activities. This second article will overview those Microsoft tools, explaining where they fit within each activity.

Microsoft's No-Cost Migration Tools

You've probably noticed the underlying theme behind the last article's discussion on best practices. There, the use of *integrated solutions* was highlighted repeatedly as a best practice for migration success. The focus on integrated solutions is critical due to the need for data sharing between each of the seven activities. Lacking a unified repository, individual tools require manual efforts that reduce your efficiency and result in less than desired results.

That repository is a key omission in Microsoft's free toolsets. Spanning over a half-dozen point solutions, Microsoft's acronym alphabet soup can be challenging to integrate successfully. Let's take a look at the solutions, and the pile of acronyms you'll need to learn. Each of these can be downloaded from the Microsoft Web site.

MSIA for Analyzing Inventory

The Microsoft Software Inventory Analyzer (MSIA) is a free tool that remotely interrogates computers throughout your Windows domain. This tool scans identified computers for known software installations, with the goal of reporting those installations in a Microsoft Excel spreadsheet.

MSIA is currently limited to scanning environments of 250 or fewer computers, and generates reports in HTML, XLS, and text only. Further, MSIA will only report on Microsoft products and applications. Products from other vendors as well as software updates and hotfixes are not enumerated.

This tool provides a reasonable level of information; however, only if your environment makes exclusive use of Microsoft applications. Third-party applications must be inventoried through a different and completely separate solution. Also, this tool's text-based reports do not store their information in a location that can be easily used by other tools.

ACT for Verifying Application Compatibility

As mentioned in the first article of this series, some applications simply will not run atop Windows 7. Others will run, but with reduced functionality or only after special modification. This application incompatibility occurs predominantly due to the dramatic kernel-level security changes that were first introduced in Windows Vista.

Microsoft's Application Compatibility Toolkit (ACT) is comprised of a set of tools that assist administrators with analyzing, classifying, and resolving just these incompatibilities. Its tools include a compatibility manager and a standard user analyzer as well as analysis tools for tracking the most common problems. Administrators using ACT begin the process by creating and deploying Data Collection Packages (DCP), which collect information about applications along with potential compatibility issues. Once DCPs have been later collected and analyzed, applications can be prioritized by business need and assessment rating while workarounds are created.

ACT includes a built-in set of pre-tested applications; however, this list is by no means complete. This tool can be considered a workbench for application-testing activities, providing a database where information about business applications can be consolidated. As with MSIA, the information gathered through this tool is not stored in a format that is used by the other tools, creating yet another silo of data for a migration project.

WAIK, ImageX, and DISM for Creating and Managing OS Instances

Actually creating OS images and scripts is accomplished using the Windows Automated Installation Kit (WAIK). This large download includes the necessary components to create and customize OS instances for later deployment.

Microsoft's primary tool for image collection is the command-line tool called ImageX. This tool captures images from reference computers and can be used for single-instance image deployment. Captured images are stored in the Windows Imaging (WIM) file format, which is a file-based disk format that was first introduced with Windows Vista. ImageX also has the ability to mount existing images for command-line modification of files and folders.

Microsoft's solution for scripted installations is the Deployment Imaging Servicing and Management (DISM) tool. This tool, a replacement for the previous Windows System Image Manager (WSIM), provides a workbench for customizing a default installation. DISM, while easier to use than WSIM, still requires advanced knowledge of during-the-install configuration requirements, and as such can be challenging for inexperienced administrators. The output from a DISM customization activity is an XML-based unattended installation script that is used in conjunction with a default installation.

USMT for Migrating User Personality

Another component of the WAIK is the User State Migration Toolkit (USMT). This command-line solution migrates user files and settings during a Windows OS deployment. USMT leverages two command-line executables—ScanState and LoadState—along with a series of XML-based configuration files to gather user data and ultimately redeploy it onto a replacement OS instance.

Although USMT's ScanState and LoadState tools are effective for handling the automated transfer of user data, these tools must be integrated into another OS deployment solution to be fully automated. This integration process typically involves the manual addition of their executables to that solution's pre- and post-deployment set of actions. Effectively using these tools requires the manual configuration of a series of XML files that contain information about the data to be transferred.

WDS for Automating OS Distribution

In another separation of tool capabilities, Microsoft's Windows Deployment Services (WDS) provides the infrastructure for actually distributing OSs to target machines. This tool ingests the scripted customizations that were created using the WAIK'S DISM, combining a DISM XML file with a default WIM to create a customized installation. WDS can also host custom WIM images that were captured via ImageX.

WDS supports the distribution of images to single computers through unicast or groups of computers through multicasting. It is installed as a role atop an instance of Windows Server 2008. As with each of the other tools thus far, configurations within WDS must be created manually. WDS image deployment can occur through the use of hardware Preboot Execution Environment (PXE) extensions or through media deployment. WDS does not support the hands-free deployment of OS images, requiring an action at each target computer to invoke an upgrade.

GPSI and WSUS for Distributing Applications and Updates

Although not technically deployment tools, Microsoft's solutions for distributing applications and updates arrive through the combination of Group Policy Software Installation (GPSI) and Windows Server Update Services (WSUS). Making use of GPSI for automated application deployment requires the creation of specially-configured "packages" of software, which are set to run silently. GPSI lacks centralized success/failure reporting capabilities, and has relatively limited targeting capabilities for discretely defining where software should be installed.

WSUS is yet another installable role within Windows Server 2008. This solution is used to install Microsoft updates and hotfixes as well as an extremely limited set of third-party drivers. WSUS does not support the automated installation of third-party updates through its graphical user interface (GUI); however, custom update installation can leverage its infrastructure through the use of custom scripts.

MDT for Project and Status Reporting

Microsoft's solution for larger-scale projects is the Microsoft Deployment Toolkit (MDT). This solution accelerator integrates limited project and status reporting with the services of the WAIK (including its subcomponents), the ACT, and MSXML 6.0. These extra services must be downloaded and installed separately from MDT.

MDT augments these individual tools through the creation of a Deployment Workbench, within which deployment-related actions are created and invoked. MDT supports two types of installations named Light-Touch Installation (LTI) and Zero-Touch Installation (ZTI). LTI installations require administrator actions to invoke an upgrade at each target computer. ZTI installations can be centrally deployed to target computers without endpoint attention but require the services of a System Center Configuration Manager (SCCM) infrastructure.

Integration Comes at a Cost. Integration Saves Cost.

Microsoft's no-cost solutions can indeed deploy OS instances to hardware. However, as you can see in this article, the savings in software cost comes without built-in automation and with an associated learning curve. With even Microsoft's solutions, if you desire an integrated approach, you'll need to invest in for-cost infrastructure management solutions such as SCCM.

Tying all these disparate activities together is the data. That data describes the as-is environment. It predicts the success for the to-be environment. And it provides the configuration characteristics that are absolutely required in order for the entire project to successfully replicate the users' experience atop the new OS. That's the theme behind this series' third and final article. That article will explore what an integrated solution looks like, managing data throughout the project life cycle to ensure migration success.