

## Very good read from the Global IT Community Association

### Chapter 1: ☐ “Choosing the Path to Windows 7”

☐

A successful deployment of a desktop operating system begins long before the first client machines are touched. The collection of information about your client machines forms the basis for creating a successful deployment plan. In this first of three posts we'll cover the first steps and the tools available to you as you start on the road to Windows 7 deployment.

The first step on the road to [Choosing a Deployment Strategy](#) is to gather the information needed to make informed decisions. Regardless of whether you are dealing with hundreds of client machines or just helping a friend, knowing the current environment is critical. Some of the basic information includes

- Number of computers to deploy Windows 7 to.
- What version of Windows do you currently have installed?
- What hardware is in these machines?
- ...and finally what applications do you use.

Depending on the size of the organization some of these questions could be tricky to answer. If you are just working on one machine or a handful of machines you can use the Windows 7 Upgrade Advisor; this will do most of the work for you, but it is not practical beyond just a few machines since you have to install it and run it on each machine individually. When you have hundreds of clients you need something a little more powerful and easier to run without much intervention. One tool is the [Microsoft Assessment and Planning Toolkit](#) . This solution accelerator that can be used to generate this inventory of assets for you, another is the [Application Compatibility Toolkit \(ACT\)](#)

, which we will come onto next.

Once you have a full understanding of what is in your organization you can then plan the deployment process. The asset list will help you determine which machines can run Windows 7 with none or minimal hardware updates and which machines will be unable to run Windows 7 and therefore require replacement.

Regardless of whether you plan to do a clean installation or an upgrade, the applications run that on these machines will need to be checked for compatibility. Application compatibility is always one of the top challenges organization face when changing a desktop OS. To help, there is the [Application Compatibility Toolkit \(ACT\)](#) - this article talks in more detail about the toolkit and how to use it. The ACT is a vital part of a deployment process, it can detect the applications running on client machines, and as mentioned above it also has the ability to report on hardware and devices that it finds on client machines. It provides you with a comprehensive list of what is out there, and don't be surprised to be surprised about what you find. Getting this view of your environment is a major step towards a successful deployment. Once armed with this information then comes the real fun, rationalization. You will have to look through all the applications on your list to determine if there is duplication, you could easily find there are 4 or 5 different programs just to read the same file format, then you need to decide which one(s) work with Windows 7 and then really which one to standardize on. The more thorough you are here could mean the difference between testing a 100 applications or testing a 1000 applications.

After the rationalization, that is not the end of the application story, even with say 100 applications each one has to be checked for compatibility with Windows 7. This may be as easy and looking on the ISVs site to see the compatibility information. You may also be faced with in-house applications that will need testing or modifications. Your deployment plan will then need to include the teams responsible for those applications so they can schedule time to work on them. You may also have applications that [require you to manually try them](#) . Some applications can have

[compatibility fixes &ndash; shims](#)

– applied to make them work. A large number of applications can be made to work very quickly and easily using shims, for example making an application think it’s running as an administrator when it’s not or that it’s running on Windows XP and has IE 6 installed. For those applications that the compatibility fixes do not work on, you may need to employ a virtualization technology such as using Virtual PC and running Windows XP Mode, using App-V or MED-V, maybe even using Terminal Services technologies. As mentioned before, there are ways to get most applications that are currently running in your environment to run while using Windows 7. The time, effort and cost to make that happen will govern the

[path you take](#)

Applications play a big part in the deployment story, even in an ideal world where all you applications run on Windows 7; you need to consider how to deploy them with your images. In the next post we’ll cover images and the tools for creating and deployment them.

---

**Twitter, etc. Sound bites:**

1. [Choosing a Windows 7 Deployment Strategy](#)

Looking to start your deployment of Windows 7? Not sure where to start or the resources available? Read this concise article on the recommended deployment strategies and the tools that support them.

---

2. [Analysing your environment with Microsoft Assessment and Planning Toolkit](#)

Do you want to know what is running in organization so you can plan a Windows 7 deployment? The Microsoft Assessment and Planning Toolkit is a free solution accelerator that can inventory your infrastructure.

---

3. [Prepare your Applications for Windows 7](#)

A key consideration when moving to Windows 7 is whether your applications will run successfully. Download the Microsoft Application Compatibility Toolkit (ACT) to access the necessary tools and documentation to evaluate and mitigate application compatibility issues before deploying Windows 7

---

4. [Five Steps to Windows 7 Application Readiness](#)

Trying to decide if the applications you use in your organization will run on Windows 7? Follow these 5 steps to determine application Readiness.

---

5. [Application Management and Preparing for a Windows 7 Deployments](#)

Read this concise article that will walk you through the variety of approaches to addressing compatibility issues and the tools available to help you.

--

6. [Commonly-used application shims here.](#)

Watch this video and see how to apply commonly-used shims to legacy application to enable them to work on Windows 7.

--

7. [Understanding Application Compatibility](#)

Why might your application not work on Windows 7? There are a few reasons such as enhanced security or retired components. This article walks you through the areas that might affect your applications.

==

**Chapter 2: “Building Windows 7 Images”**

In the previous post we looked at the key information and first steps required to perform a successful deployment of Windows 7, we looked in some detail at one of the main concerns organizations have when deploying a new OS, application compatibility. In this post we'll look at the resources available to help prepare for the actual deployment of Windows 7.

Efficient deployment of a Windows OS to many different machines usually involves using an image. Until very recently that image was a sector-based image and organizations usually had one for each type of client hardware they own.

Today we have file-based images in the Windows Imaging Format (WIM). This format offers a number of advantages over sector-based images such as being hardware agnostic within processor architecture, e.g. you will need separate images for x86 and x64 processors. WIMs are usually smaller than their sector-based image equivalent, easier to maintain and patch, you don't need hundreds of them to support your client hardware base and they allow for more flexible deployment options. To go along with this new image format comes a slew of new tools and documentation to help create and maintain them. The main tool is the [Windows Automated Installation Kit \(AIK\) for Windows 7](#).

I called it a tool; in fact it's a suite of tools and documentation to help with the image creation and maintenance.

The one thing that hasn't changed over the years is the concept; regardless of whether you use sector-based images or file based images you do start with a reference machine, prepare it for capture and then capture it. What has changed is the way you do this and the strategy you follow. In the article [Choosing and Image Strategy and Building Windows 7 System Images](#) ,

the 3 primary strategies for imaging are discussed. In brief these are “Thick”, “Thin” and “Hybrid”.

A “Thick” image is one that contains the OS and all applications you want to have available as soon as the imaging process is complete. As the name suggests it’s the bigger of the imaging strategies.

A “Thin” image is effectively the opposite of “Thick”, it contains the very basic information, and other items like the applications are handled at deployment time.

Finally “Hybrid” is a combination of the other two, core applications people need to be able to use right away are installed, and others are handled at deployment time or later.

Which one to use depends on your requirements, again either way the tools to create the images for the three strategies are the same. The core tools are Windows PE, SysPrep, ImageX, and DSIM (Deployment Image Servicing and Management). These tools – in order - allow you to boot a machine to install Windows 7, prepare it for capture and deployment, capture the image ready for deployment and then subsequently maintain it. I could write about the process, even point you at the training kit for Configuring Windows 7, (Imaging is approximately 13% of exam 70-680), but it’s better to see it in action, so first here is a video of [Sysprep and ImageX being used to generalize and capture a custom](#) and a video of [DSIM servicing an offline mounted Windows 7 image](#)

The creation and maintenance of images these days is pretty straightforward and certainly a lot more efficient. If you are not using file-based Windows Image format (WIM), downloading the WAIK documentation will help you in switching to this deployment method. Once you have your images ready, the next step is to get them onto the clients. In the final post we will look at ways to get the image file onto a client machine.