Altiris
Deployment Solution 6.5

Standardizing Desktop images

How to create a generic image and use Windows PE for automatic HAL detection and utilization

Created by: Peter Giesa – UBM Professional Services
18.11.2005

Based on the Altiris document:
Altiris Standardizing Desktop images,
Prepared By: Steve Stoddard – IS Altiris Technology Manager

Executive Summary

When setting up new computers as part of a new role out or migration, one of the quickest ways to setup and configure the computers is accomplished using computer imaging.

Deploying an image of a computer without properly preparing it can cause problems and even fail if the hardware of the source computer does not match the destination hardware. Creating a catalog of different images to span all the hardware in your organization is a daunting task. To reduce the amount of time managing images, it is crucial to learn how to create a truly hardware independent image.

This document will outline the steps needed to create a hardware independent image that will span desktops and laptops at your organization. The scope of this document speaks only to desktops and laptops and focuses on using Windows XP as the OS for the base of the image.

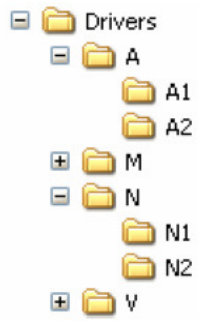
Image Obstacles

There are three major obstacles to overcome when creating a standard hardware independent image:

1. Supplying 3rd party drivers for all of the different hardware configurations.
2. Mass Storage Controller Support.
3. Determining the proper HAL.

Supplying 3rd Party Drivers

You may notice that with newer machines there are fewer and fewer drivers that are built into the operating system. There are several factors that contribute to this, but the bottom line is that we need to be able to tell the operating system where the drivers are for these newer hardware devices. One of the ways that you can do this is to include a drivers folder in your image that has all of the drivers needed by your organization. It is recommended that you group your drivers in a way that makes the most sense to you. The illustration below gives an example of how to do this.



The example above uses short abbreviations for common driver category names i.e. A = Audio, M = Modem, N = Network, V = Video. The reason for abbreviations or short file names is because we will be adding each path as a unique path the OemPnpDriversPath

in the sysprep.inf file. This entry currently has a **limitation of 4096 characters** so as you can see the more different paths you have the closer you may get to this limitation.

The example above is only a suggestion and you can name your folders differently. The A1, A2, etc folders are simply placeholders wherein you can place specific driver files for the appropriate sections.
You can create your own name specific folders, but it is recommended that you leave a few folders empty as placeholders so if you wish to add driver files into the image at a later date you can do so. The reason for the placeholders is that when sysprep runs it updates the
HKLM\software\Microsoft\Windows\CurrentVersion\DevicePath path in the registry with all of the specific paths from the OemPnpDriversPath in the sysprep.inf file. Now when mini setup runs, after having deployed the image, it will look in this key for all possible paths to driver files. By having some placeholders you will not have to rerun sysprep on your reference computer to recreate the entries. Instead you can just edit the image file and drop in more driver files.

Once you have created your Drivers folder all you'll need to do is add an entry into the OemPnpDriversPath section of your sysprep.inf file.

Example of the OemPnpDriversPath section

```
OemPnpDriversPath="\Drivers\A\A1;Drivers\A\A2\Drivers\M;\Drivers\N\N1;
\Drivers\V\V1;"
```

Sysprep knows to append the %systemdrive% value to the front of each of the entries. Unfortunately you can't just specify a parent folder and have mini setup be able to drill down through it to find the proper driver. You must specify each path in this section hence the reason to try and keep the structure as short and flat as possible so that you will not exceed the 4096 character limitation.

In order for sysprep to properly load the Oem Drivers you'll need **at least the following sections in your sysprep.inf file**:

[Unattended]
DriverSigningPolicy=Ignore
OemPreinstall=Yes
OemPnPDriversPath="\Drivers\A\A1;\Driver\A\A2;"…

Mass Storage Controller Support

An issue early on in the attempts to create a standard desktop image was that of providing support for all of the different mass storage controllers. If you had the incorrect driver your system was either unstable or even failed to boot into the OS. Microsoft has helped solve this issue with the inclusion of the SysprepMassStorage section in the sysprep.inf file. You can either manually place all of the mass storage drivers to this section or even better have sysprep build it for you. By adding a *BuildMassStorageSection=Yes* to the sysprep.inf file and then running *sysprep.exe –bmsd* command, sysprep will automatically create all of the entries in the *SysprepMassStorage* section with all of the mass storage drivers included in the OS.

Along with specifying the *SysprepMassStorage* and *BuildMassStorageSections* in your sysprep.inf file, you will also need to add a cmdlines.txt file containing the following data:

[Commands]
"C:\Sysprep\sysprep.exe -clean"

(Note: If you use setupmgr to build your sysprep.inf file it will automatically create the structure and add the entry as long as you specify in setupmgr to run this command: *c:\Sysprep\sysprep.exe –clean*.)

The cmdlines.txt file will be stored in the C:\Sysprep\i386\OEM\ folder.

HALs

HAL Defined

A hardware abstraction layer is a component of an operating system that functions something like an API. (In strict technical architecture, hardware abstraction layers reside at the device level, a layer below the standard API level.) It allows programmers to write applications and game titles with all the device-independent advantages of writing to an API, but without the large processing overhead that APIs normally demand.

In order for an image to be successfully deployed to a machine the HAL must be compatible. If you build your image on a machine that uses the ACPI Uniprocessor PC HAL then that image will work fine on all computers that can properly negotiate this HAL. However, if you try to put this image on a machine that can only use the Advanced Configuration and Power Interface HAL then the OS will fail to load.

To truly make a hardware independent image that can span all of your client computing hardware you'll need to make certain that you have properly placed the right HAL into the image. Most all of today's desktops and laptops use one of three HALs:

- Advanced Configuration and Power Interface HAL halacpi.dll
- ACPI Uniprocessor PC HAL halaacpi.dll
- ACPI Multiprocessor PC HAL halmacpi.dll

These different HAL files are renamed to hal.dll and placed in *%systemroot%\system32\HAL.dll*. You can't have multiple HALs configured in your image so you'll need to make this determination post image deployment, but prior to booting into the newly deployed image.

To determine which HAL the machine uses go to Device Manager and under the Computer Node you can see which HAL is being used. The driver that the HAL reference is always labeled as HAL.dll in the *%systemroot%\system32* folder, but if you view the properties of this file you can see what the Internal Name is.

To place the HAL into the image you can use the FIRM utility that ships with Altiris Deployment Solution. This utility allows you to copy files into partitions that the environment has yet to assign a drive letter or recognize yet.

Example: At the root of the drivers folder I would have two files: halacpi.dll and halaacpi.dll. I would deploy my corporate image and then use FIRM to copy in the proper HAL.dll i.e.

Firm.exe copy prod:drivers\halacpi.dll prod:windows\system32\HAL.dll.

Beside this, the different HALs can also reside in the express-share in a directory called HAL, which makes the process more flexible, if new HAL types need to be supported in the future.

(For more information on FIRM and its uses see the product documentation for Deployment Solution)

For a sample of how to use FIRM within a batch process to copy in the proper HAL after image deployment see Appendix B.

Creating the Image

Prerequisites

The following items are prerequisites:

- Sysprep from a Windows 2003 Server CD-Rom. This is the best version of sysprep to use and is compatible with Windows XP.
- Familiarity with Sysprep.
- Windows XP Volume license CD with corresponding key.
- Imaging technology.
- A folder of all driver files needed to install all of the various drivers for the varying hardware found throughout the organization.
- HALs – You will need a copy of the Advanced Configuration and Power Interface Hal (halacpi.dll) and the ACPI Uniprocessor Hal (halaacpi.dll) and any other HAL files that may be specific to your desktops.

Creating the image - Steps

Obtain a reference computer that can be used now and in the future for creating your image.

- Install and configure the operating system.
- Install corporate applications.
- Clean up.
- Prepare for deployment.

Obtain Reference Computer

It is important when you are creating a hardware independent image that you have a reference computer that can always be used. This will make it easier when you wish to update the image with future applications etc.

Install and Configure the Operating System

You'll want to install the operating system using your volume license media and key. **It is important that you do not add the machine at this time to your corporate domain**. Sysprep will add the machine to your corporate domain if you choose, but for initial image creation leave the machine off of the domain.

Do not configure any hardware that isn't absolutely necessary. If XP was unable to plug and play your hardware leave it uninstalled. By leaving the hardware uninstalled it will make it easier to deploy the image.

If you wish to specify the local administrator password in sysprep then you will need to leave the password blank when setting up the image.

Install Corporate Applications

Create a local "install" user account and give this user local administrator privileges. Now, logon as this user and install all of the applications that you need on your image. When the time comes we will copy this users profile to the "Default User" profile which will then help guarantee that all users who use the system will have visibility into all of the installed applications.

Clean Up

After you have installed all of the applications and made any configuration changes you can now remove all temporary files and clean up the environment in preparation for imaging. Some areas to clean up are:

1. Temp Files
2. Cookies
3. Internet History
4. Empty Recycle Bin
5. Recently used programs and documents

Prepare for Deployment

1. Reboot the computer and logon as the local administrator.

2. Copy your corporate drivers folder to the root of the %systemdrive%.

3. Copy the "install" user profile to "Default User" (note: You'll need to enable hidden files and folders in order to see the Default User folder.)
   This step is need, because the default administrator profile can not be copied, while it is in use.

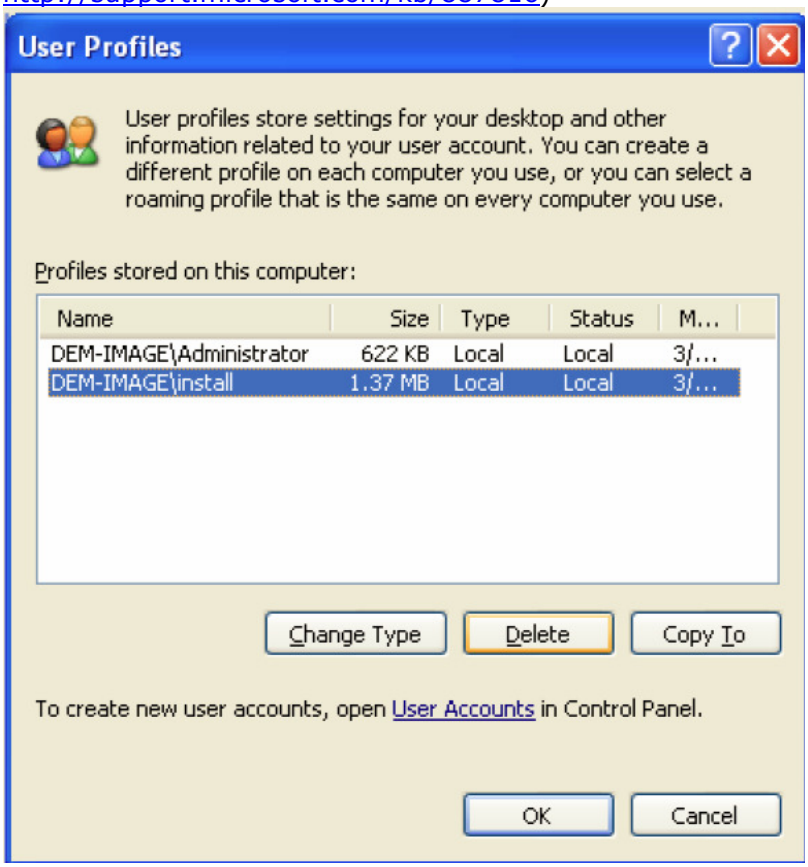
   - To copy the profile go to properties of "My Computer" and then go to the advanced tab. From here select the user profile settings.

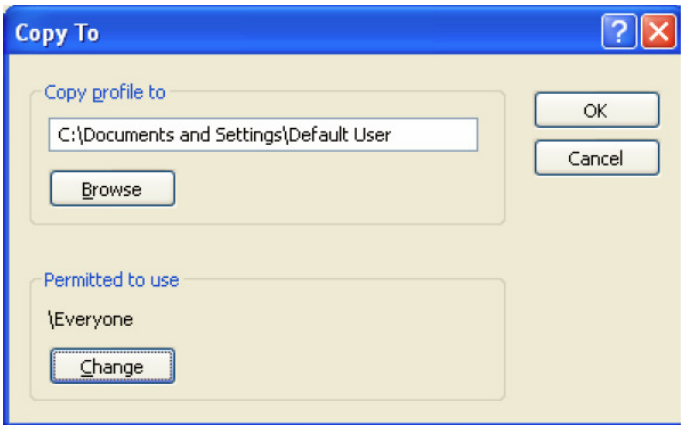
   - Now select the install profile and click on the "Copy To" button.

   - In this next dialog browse to the "Default User" folder and then in the "Permitted to use"

   - section change the group here to the local "Everyone" group. Hit OK.

   - (Note: In SP2 for XP changes to the default user profile may not take affect. Microsoft has released a KB on this with a Hot Fix that addresses the issue. http://support.microsoft.com/kb/887816)





4. Delete the "install" profile. Using the User Profile form delete the install profile.

5. Delete the "install" user account.

6. In Device Manager remove any uninstalled devices along with any devices that you may have manually configured.

7. Copy all the sysprep files into a folder on the root labeled "Sysprep". If you've already created a custom sysprep.inf file then place it in here as well.

8. If you don't have a custom sysprep.inf file then run setupmgr to create one. You'll need your Windows XP Volume License key if you wish to fully automate sysprep. See appendix A for some sample sysprep files and what yours should look similar to.
   In a later stage you are going to copy your sysprep.inf file to the express-share. The Deployment Server will use the sysprep.inf file from the express-share and has the possibility to replace specific entries with variable.

   When creating your sysprep.inf file, use the following tokens in setupmgr:

(User) Name:	%USER_NAME%
Computer Name:	%COMPNAME%
Product Key:	You can use any number, this value will be replaced with a token in the next step.

9. In the [Unattended] section define your OEM drivers path and ensure that the proper entries exist, as described above.

   Open your just created sysprep.inf file with notepad, go to [UserData] and use the following entry for the product key:
   ProductKey=%PROD_LIC%

   At the end of the file create a new section:
   [SysprepMassStorage]

10. Run sysprep.exe –bmsd to build the SysprepMassStorage section.

- In order for the SysprepMassStorage section to populate with all of the mass storage drivers that the OS knows about you'll need to make sure you have the following section in your sysprep.inf file:

  [Sysprep]
  BuildMassStorageSection=Yes

- Your sysprep folder should have a i386\oem folder in it with a file named cmdlines.txt in it. The cmdlines.txt should have this entry:

  [Commands]
  "C:\Sysprep\sysprep.exe –clean"

  The –clean will prevent Windows from trying to load drivers for nonexistent mass storage controllers

11. Copy the created sysprep.inf file from your c:\sysprep directory to the express share into the Sysprep directory. You can use different sysprep.inf files i.e. for different language settings. You can rename the file to be i.e. sysprep_fin.inf.

12. If your workstation has the NSAgent installed, make sure to follow the Altiris kb-Article on how to remove the GUID of the NSAgent, before the image is created (the same information can be found in the Notification Server Reference Guide). Do the same for any software that is part of the image and requires a unique GUID on all of the workstations.

13. Run sysprep.exe –mini –reseal to prepare the image for deployment.

    - This process can take 5 – 15 minutes to run so be patient. Once sysprep is done running the machine will shut down and be ready for you to take an image.

14. Capture an image of the reference computer using Altiris' Imaging Technology. If you use a Deployment Solution Job with a "Create Disk Image" task, make sure to ENABLE the option: "Do not boot to Windows".

Deploying the Image

Now that you have captured your image there is only one step left in making the image hardware independent. Because the image itself may not contain the proper HAL.dll file for the destination hardware you will have to evaluate this post image deployment time, but before reboot.

If you are booting into a Windows PE environment to facilitate in your image deployment then you can query the PE registry to determine which HAL was loaded for PE and the copy in the correct HAL into the destination using a utility such as FIRM. Registry value to query:
HKLM\SYSTEM\ControlSet001\Enum\Root\ACPI_HAL\0000\HardwareId. The value should either be acpiapic_up for Uniprocessor HAL type or acpipic_up for the Advanced Configuration and Power Interface HAL. See Appendix B for an example of how to use Firm to copy in the correct HAL in Win PE.

If you are not using PE then you may need to setup deployment jobs that group your destined computers into collections that are specific to the HAL that they need so a post image task would be to then again use FIRM and copy in the proper HAL.dll.

If Windows PE is not available as a pre-boot OS, Conditions can be used to target the proper HAL to the workstations via Linux or DOS pre-boot..
It is recommended to create a directory called HAL in the express share and store all needed HAL files in there (halacpi.dll, halaacpi.dll and halmacpi.dll).
In your image distribution job below the Distribute Disk Image task, create a new Run Script task and use either DOS or Linux as the execution platform.
Use the FIRM command to copy the according HAL to the workstation, after the image has been deployed.

Appendix A - Sysprep Samples

;SetupMgrTag

```
[Unattended]
 OemSkipEula=Yes
 InstallFilesPath=C:\sysprep\i386
 DriverSigningPolicy=Ignore
 OemPreinstall=Yes
 OemPnPDriversPath="\Drivers\A\A1;\Drivers\A\A2;\Drivers\V\V1;”
 ResetSourcePath=C:

[GuiUnattended]
 AdminPassword=”Sample”
 EncryptedAdminPassword=No
 AutoLogon=Yes
 AutoLogonCount=1
 OEMSkipRegional=1
 OEMDuplicatorstring="Corporate Image"
 TimeZone=10
 OemSkipWelcome=1

[UserData]
 ProductKey=[Your key goes here]
 FullName="Sample User"
 OrgName="Sample"
 ComputerName=*

[Display]
 Xresolution=1024
 YResolution=768

[SetupMgr]
 DistFolder=C:\sysprep\i386
 DistShare=windist

[Identification]
 JoinDomain=sample.com
 DomainAdmin=sample\install
 DomainAdminPassword=samplePassword

[Networking]
 InstallDefaultComponents=Yes

[Sysprep]
 BuildMassStorageSection=Yes

[SysprepMassStorage]

*pnp0a00=c:\windows\inf\machine.inf
*pnp0a01=c:\windows\inf\machine.inf
*pnp0a04=c:\windows\inf\machine.inf
*pnp0a03=c:\windows\inf\machine.inf
pci\cc_0604=c:\windows\inf\machine.inf
```

Appendix B – Programmatically Updating the HAL in PE

The following is a sample batch file that can be run within the PE Shell to determine which HAL was loaded and then copy the proper HAL from a stored location in the Drivers folder.

```
@ECHO OFF

:HAL

REM Lets query the registry to see what hal is used.

REM First hal to check for is the Advanced Configuration and Power
Interface Hal

REG QUERY HKLM\SYSTEM\ControlSet001\Enum\Root\ACPI_HAL\0000 /v HardwareId
|Find /I /C "acpipic_up"

REM Found the hal lets move down to ACPI

IF %ERRORLEVEL% EQU 0 GOTO ACPI

REM Hal not found lets check the next value

REM Second hal to check for is the Uniprocessor Hal

REG QUERY HKLM\SYSTEM\ControlSet001\Enum\Root\ACPI_HAL\0000 /v HardwareId
|Find /I /C "acpiapic_up"

REM Found the hal lets move down to AACPI

IF %ERRORLEVEL% EQU 0 GOTO AACPI

GOTO :NOHAL

:ACPI

REM Copy in the correct HAL

%SystemDrive%\Programs\Rdeploy\Firm copy prod:drivers\halacpi.dll
prod:Windows\System32\HAL.dll

GOTO END

:AACPI

REM Copy in the correct HAL

%SystemDrive%\Programs\Rdeploy\Firm copy prod:drivers\halaacpi.dll
prod:Windows\System32\HAL.dll

GOTO END

:NOHAL
```

```
ECHO Can not find the appropriate HAL to use. Leaving as is.

:END

ECHO Process finished successfully.

EXIT
```

References

HAL options after Windows XP or Windows Server 2003 Setup

http://support.microsoft.com/kb/309283

How to Determine the Hardware Abstraction Layer (HAL) That Is Used in Windows XP

http://support.microsoft.com/?kbid=298898

How to Troubleshoot Windows 2000 Hardware Abstraction Layer Issues

http://support.microsoft.com/kb/237556/EN-US/

Maximum length of the OEMPnpDriversPath value in Windows XP - 4096

http://support.microsoft.com/default.aspx?scid=kb;en-us;312535&Product=winxp

Answer file entries for the [Unattended] section

http://www.microsoft.com/resources/documentation/WindowsServ/2003/all/techref/en-us/Default.asp?url=/Resources/Documentation/windowsserv/2003/all/techref/en-us/w2k3tr_s_unattended.asp

hardware abstraction layer

http://www.cnet.com/Resources/Info/Glossary/Terms/hardwareabstractionlayer.html

Changes in behavior of the SysPrep and RIPREP tools after you install Windows XP Service Pack 2

http://support.microsoft.com/kb/887816