HP SmartStart Scripting Toolkit
Win32 Edition
Best Practices

HP SmartStart Scripting Toolkit Win32 Edition Best Practices

Contents

About This Guide

This guide describes the implementation of the HP Smart Scripting Toolkit utilities with Microsoft® Windows® Preinstallation Environment (Windows PE).

This guide **does not** provide details about installing the Microsoft Windows operating system or using Windows PE.

Audience Assumptions

This guide is intended for experienced network administrators. Some of the utilities in the Toolkit can cause a loss of data if used incorrectly. Thus, a clear understanding of the system and these utilities is critical.

Where to Go for Additional Help

For issues or problems not addressed by this guide, refer to the following resources for more information.

- The SmartStart Scripting Toolkit website: http://www.hp.com/servers/sstoolkit

- The Microsoft website: http://www.microsoft.com/

Customer Feedback

HP offers standard e-mail support for the SmartStart Scripting Toolkit. Refer to http://www.hp.com/go/support, select your particular HP ProLiant server product, and use the online form to submit your question.

Telephone Numbers

For the name of your nearest HP authorized reseller:

- In the United States, call 1-800-345-1518.

- In Canada, call 1-800-263-5868.

For general HP technical support:

- In the United States and Canada, call 1-800-HP INVENT (1-800-474-6836).

- Outside the United States and Canada, refer to http://www.hp.com/.

1

Introduction

This document describes how to best utilize the Win32 edition of the SmartStart Scripting Toolkit to configure HP ProLiant servers and contains information about using the Toolkit utilities in an unattended environment.

NOTE: This document does not include information about installing the operating system.

The best practices also includes suggestions about how to organize utilities and data files by following a set of operational procedures that standardize configuration procedures and help reduce errors.

> **CAUTION:** Because of the potential risk of data loss, be sure that all necessary precautions are taken so that mission-critical systems are not disrupted if a failure occurs.

Toolkit Changes

Previous versions of the SmartStart Scripting Toolkit utilities were designed for the MS-DOS environment. However, limitations in MS-DOS have become a significant problem as hardware and software have evolved. In response to this issue, HP has migrated the SmartStart Scripting Toolkit to a Win32 environment to provide better scripting and better hardware support using Win32 drivers and to take advantage of built-in Win32 features, such as integrated network support.

The Win32 edition of the Toolkit provides the same functionality as the earlier MS-DOS version. However, tools, arguments, and data files are different in the Win32 edition of the Toolkit, so you must update your customized scripts for working in a Win32 environment. Data files have been migrated to industry-standard XML format to provide improved extensibility. For more information about the Toolkit utilities, refer to the *HP SmartStart Scripting Toolkit Linux and Win32 Editions User Guide*.

Minimum Requirements

Before beginning the deployment process, be sure to have the following items available:

- *HP SmartStart Scripting Toolkit Linux and Win32 Editions User Guide*

- SmartStart Scripting Toolkit Win32 Edition

- Microsoft Windows PE Build Tools 2004

- Microsoft Windows XP, Service Pack 2 (SP2)

2

Microsoft Windows Preinstallation Environment

Microsoft Windows Preinstallation Environment 2004 (Windows PE 2004) is a small footprint of the Microsoft Windows XP environment that replaces MS-DOS and enables you to run tools in a 32-bit Windows environment. However, Windows PE 2004 is based on the Windows XP SP2 kernel running in protection mode. Therefore, only a subset of features and application program interfaces (APIs) are available in this environment. This document is applicable to Windows PE 2004 only. It is not backward compatible with previous versions of Windows PE.

Windows PE enables you to install the Windows operating system, establish a connection with network servers, and perform hardware configuration using the SmartStart Scripting Toolkit. You can use Windows PE to customize and configure your servers before the operating system is installed.

The Microsoft Windows OEM Preinstall Kit (OPK) CD contains a bootable image of Windows PE. The OPK also contains tools, scripts, and documentation that enable you to create and customize your own Windows PE image. Customers are required to have the Microsoft Enterprise Agreement (EA) or Software Assurance Membership (SAM) to gain access to the OPK or Windows PE. For more information about the OPK, refer to the Microsoft website at <ins>http://www.microsoft.com/</ins>.

Windows PE Build Process

The OPK contains a collection of scripts and tools to help you build your own Windows PE image. You must customize your Windows PE image to include the latest HP drivers and the SmartStart Scripting Toolkit utilities, data files, and customized scripts.

NOTE: While the latest ProLiant drivers are always available on the HP support website (<ins>http://www.hp.com/support/files</ins>), unpackaged ProLiant drivers can also be obtained from the SmartStart CD at Compaq\Drivers\Wnet.

The following sections, "Building a Windows PE Base Image" and "Customizing Windows PE with HP Drivers, Data Files, and Toolkit Utilities," give an overview of the Windows PE build process with HP drivers and tools and should be executed in the Windows command prompt (CMD.EXE) environment.

Building a Windows PE Base Image

This section describes how to build a Windows PE base image. The base image can then be customized as described in the next section. For detailed information about customizing a Windows PE base image, refer to the Windows PE documentation.

To build a Windows PE base image:

1. Copy the Windows PE Build Tools and files for the appropriate target server platform to a temporary directory:

    a. Create the temporary directory:

        ```
        mkdir WinPEBuildTools
        ```

    b. Copy files from the OPK to the temporary directory:

        ```
        copy PathToWinPEFiles\WinPE\*.* WinPEBuildTools
        ```

        ```
        copy PathToWinPEFiles\Tools\%Platform%\*.* WinPEBuildTools
        ```

2. Build the folder structure of Windows PE image with Plug-N-Play (PnP) and Windows Management Instrumentation (WMI) support enabled:

    ```
    Mkimg.cmd PathToXPFiles WinPE_Directory /PnP /WMI
    ```

3. Build Windows PE optional components:

    a. Build the components:

        ```
        Buildoptionalcomponents.vbs /s:PathToXPFiles
        /d:OptionalComponentDirectory /ADO /HTA /WSH /X
        ```

    NOTE: Optional components include support for Windows Script Host (WSH), HTML Applications (HTA), and ActiveX Data Objects (ADO).

    b. Copy the optional component files to the Windows PE folder:

        ```
        xcopy /e OptionalComponentDirectory WinPE_Directory
        ```

    c. Edit `startnet.cmd` to include the string `oc.bat`:

        ```
        echo oc.bat >> WinPE_Directory\Platform\system32\startnet.cmd
        ```

Customizing Windows PE with HP Drivers, Data Files, and Toolkit Utilities

NOTE: Tools and data files are not required during the boot process, so they do not need to be stored on the Windows PE CD. These files can reside on a diskette, a USB key drive, or a network share.

Windows PE contains a set of drivers and other files imported from Windows XP that are critical during the boot process of Windows PE. In addition to the drivers imported from Windows XP, additional Windows Driver Model (WDM) compliance drivers can be added to the Windows PE image during customization. HP ProLiant drivers **must** be added during Windows PE customization to ensure the Toolkit utilities function properly. Microsoft OPK contains tools to help you update drivers to the Windows PE image.

The following table lists HP ProLiant software required to ensure that the Toolkit utilities have full access to the hardware.

Table 2-1: Required Software

HP ProLiant Software	Version
Network interface controller (NIC) drivers	Windows Server 2003
Storage controller drivers	Windows Server 2003
Health drivers (iLO Advanced and Enhanced System Management Controller Driver and Advanced System Management Controller Driver)	Windows 2000
Lights-Out interface drivers (iLO Management Interface Driver and RILOEII Board Driver)	Windows 2000, Windows Server 2003
SM2USER.DLL from the HP Insight Management Agent	Windows 2000, Windows Server 2003

The Windows PE image can be customized in either of the following ways:

- **Customizing manually** requires that you obtain the latest drivers from the HP support website and install them into the Windows PE image manually.

- **Customizing from a Toolkit script file** simplifies the process and requires minimal assistance. A script file, which is provided with the Toolkit, automatically copies the drivers to the appropriate folder and registers them with Windows PE.

The following sections describe each of these methods in more detail.

Customizing the Windows PE Image Manually

To customize the Windows PE image manually, download the latest network, storage, Lights-Out interface, and Advanced Management Controller drivers from HP, and then update the Windows PE image with the new drivers.

NOTE: In the initial release of the Win32 edition of the SmartStart Scripting Toolkit, the drivers required to customize Windows PE are also available in the HP Drivers directory of the Toolkit package.

NOTE: To ensure full access to the hardware, be sure to install the latest HP ProLiant drivers.

To obtain drivers from HP:

1. Download the latest drivers from http://www.hp.com/support/files.

2. Execute the downloaded Smart Component installer.

3. Select **Extract** to extract the driver binary and setup INF files from the HP ProLiant Package Setup.

4. Select a destination for the extracted files.

5. Close the HP ProLiant Package Setup.

To customize Windows PE with new driver updates, based on the image built in the previous section, "Building a Windows PE Base Image," choose one of the following methods.

Non-Mass-Storage Drivers

1. Update the Windows PE image with new device drivers by running the following command for each driver extracted from the HP ProLiant Package Setup:

    ```
    Drvinst.exe /inf:PathToDriver WinPE_Directory
    ```

2. Copy Toolkit utilities, data files, and other files to the Windows PE directory.

3. Build an ISO CD bootable image of Windows PE from the Windows PE directory:

    ```
    oscdimg.exe -n -betfsboot.com WinPE_Directory WinPE.iso
    ```

4. Using the ISO image file and CD creator software, build the Windows PE CD, and verify that the CD is working properly.

Mass Storage Drivers

1. Update the Windows PE image with new storage device drivers:

    a. Create a new directory:

    ```
    mkdir WinPE_Directory\Platform\System32\StorageDriver
    ```

    b. Copy the driver files to the new directory:

    ```
    copy PathToStorageDriverFiles\*.*
    WinPE_Directory\Platform\System32\StorageDriver
    ```

    c. Update the **[OemDriverParams]** section in the *WinPE_Directory\Platform*\System32\Winpeoem.sif file to the following:

    ```
    [OemDriverParams]

    OemDriverRoot=""

    OemDriverDirs=StorageDriver
    ```

    d. If adding CPQCISSM or MEGAIDE drivers to the Windows PE image, make multiple copies of the driver folder, and modify the txtsetup.oem file as described in the following section, "CPQCISSM and MEGAIDE Drivers."

    For more details about customizing Windows PE, refer to the OPK documentation.

    NOTE: You can also use the F6 option to instruct Windows PE to load mass-storage drivers from a diskette while Windows PE is booting. Using the F6 option has the same result as loading mass-storage drivers during Windows setup.

2. Copy Toolkit utilities, data files, and other files to the Windows PE directory.

3. Build an ISO CD bootable image of Windows PE from the Windows PE directory:

    ```
    oscdimg.exe -n -betfsboot.com WinPE_Directory WinPE.iso
    ```

4. Using the ISO image file and CD creator software, build the Windows PE CD, and verify that the CD is working properly.

CPQCISSM and MEGAIDE Drivers

The CPQCISSM driver is required for Windows PE for support of many new ProLiant servers. If adding CPQCISSM or MEGAIDE drivers to the Windows PE image, make three copies of the driver folder, and modify the txtsetup.oem file to target the driver specifically for the storage controller. These drivers each support several different storage controllers and therefore require a separate folder for each driver that targets a specific storage controller.

1. Make three copies of the driver.

   ```
   Xcopy /i WinPE_Directory\Platform\System32\Cpqcissm
      WinPE_Directory\Platform\System32\Cpqcissm2
   Xcopy /i WinPE_Directory\Platform\System32\Cpqcissm
      WinPE_Directory\Platform\System32\Cpqcissm3
   ```

2. Edit the Defaults section of the txtsetup.oem file so that each folder targets a different storage controller. The list of supported controllers is located under the SCSI section of the txtsetup.oem file. The following are examples of each txtsetup.oem.

   WinPE_Directory\Platform\System32\Cpqcissm\Txtsetup.oem contains:

   ```
   [Defaults]
   SCSI = B060
   ```

   WinPE_Directory\Platform\System32\Cpqcissm2\Txtsetup.oem contains:

   ```
   [Defaults]
   SCSI = B178
   ```

   WinPE_Directory\Platform\System32\Cpqcissm3\Txtsetup.oem contains:

   ```
   [Defaults]
   SCSI = 0046
   ```

3. Be sure the new driver folders are registered with *WinPE_Directory\Platform*\System32\Winpeoem.sif:

   ```
   OemDriverDirs=cpqcissm, cpqcissm2, cpqcissm3, etc…
   ```

Repeat steps 1 through 3 for the MEGAIDE driver if necessary. Set the value of the Default section according to the SCSI section.

Customizing the Windows PE Image Using a Toolkit Script File

To simplify the process of customizing the Windows PE image with the latest ProLiant drivers, the Win32 edition of the Toolkit provides a script file, hp_WinPE_customize.cmd, that automatically registers and copies the drivers to the appropriate folder. User interaction is still required to set up the Windows command prompt (CMD.EXE) environment.

NOTE: To ensure full access to the hardware, be sure to install the latest HP ProLiant drivers.

The script uses drivers shipped with the SmartStart Scripting Toolkit to customize the Windows PE image. Before running the script, the following environment variables must be set appropriately.

NOTE: The script must be run after the Windows PE base image is built but before other modifications are made to the Windows PE image.

Table 2-2: Environment Variable Settings

Variable	Command Example	Description
`HPDriversDir`	`set HPDriversDir=D:` `\FlatDrivers`	This variable sets the directory where the Toolkit packaged driver files are stored.
`WinPEBuildToolsDir`	`set WinPEBuildToolsDir=` `D:\WinPEBuildTools`	This variable sets the directory where the Windows PE Build Tools are stored.
`WinPEDestDir`	`set WinPEDestDir=D:\WinPE`	This variable sets the directory where the Windows PE base image was built.

To customize Windows PE with new driver updates, based on the image built in the previous section, "Building a Windows PE Base Image":

1. Set up the environment variables:

    ```
    set HPDriversDir=ToolkitDrivers

    set WinPEBuildToolsDir=WinPEBuildTools

    set WinPEDestDir=WinPE_Directory
    ```

2. Update the Windows PE image with new device drivers by executing the hp_WinPE_customize.cmd script:

    ```
    hp_WinPE_customize.cmd
    ```

3. Copy Toolkit utilities, data files, and other files to the Windows PE directory.

4. Build an ISO CD bootable image of Windows PE from the Windows PE directory:

    ```
    oscdimg.exe -n -betfsboot.com WinPE_Directory WinPE.iso
    ```

5. Using the ISO image file and CD creator software, build the Windows PE CD, and verify that the CD is working properly.

Customizing Windows PE Examples

This section provides two examples of how to build a customized Windows PE image. These are examples only. Additional customization might be necessary for your environment.

The examples assume:

*   The target Windows PE is for a 32-bit x86 system.

*   The Microsoft Windows XP SP2 CD is in the F: drive.

*   The HP SmartStart CD is in the G: drive.

*   The Windows PE target destination directory is D:\WinPE.

*   The Windows PE 2004 Build Tools directory is D:\WinPEBuildTools.

*   The Windows PE temporary optional components directory is D:\WinPEBUildTools\OptionalComponents.

- HP Smart Components have been downloaded and extracted to:
  - D:\hp\Storage for storage controller drivers
  - D:\hp\Network for network controller drivers
  - D:\hp\SysMgmt for the HP Advanced System Management Controller drivers (with and without iLO support)
  - D:\hp\LO for Lights-Out Interface drivers (iLO and RILOE II)
  - D:\hp\AgentDLL for SM2USER.DLL from the Insight Management Agent
- HP SmartStart Scripting Toolkit utilities have been extracted to D:\hp\SSTK.
- The final Windows PE ISO image is stored as D:\WinPE.ISO.

Customizing the Windows PE Image Manually

To build a customized Windows PE image manually:

1. Build the Windows PE image:

   ```
   D:\WinPEBuildTools\mkimg.cmd F: D:\WinPE /PnP /WMI
   ```

2. Build Windows PE optional components if necessary:

   a. Build the components:

   ```
   D:\WinPEBuildTools\Buildoptionalcomponents.vbs /s:F: /d:
   D:\WinPEBuildTools\OptionalComponents /ADO /HTA /WSH /X
   ```

   b. Copy the optional components into the Windows PE directory:

   ```
   xcopy /e D:\OPKTools\OptionalComponents D:\WinPE
   ```

   c. Edit startnet.cmd so that oc.bat is launched at the end of the boot process:

   ```
   echo oc.bat >> D:\WinPE\i386\system32\startnet.cmd
   ```

3. Update the Windows PE image with HP network device drivers:

   ```
   D:\WinPEBuildTools\Drvinst.exe /inf:D:\hp\Network D:\WinPE
   ```

4. Update the Windows PE image with HP Advanced System Management Controller drivers:

   ```
   D:\WinPEBuildTools\Drvinst.exe /inf:D:\hp\SysMgmt D:\WinPE
   ```

5. Update the Windows PE image with Lights-Out interface drivers:

   ```
   D:\WinPEBuildTools\Drvinst.exe /inf:D:\hp\LO D:\WinPE
   ```

6. Update the Windows PE image with the SM2USER.DLL file from the Insight Management Agent:

   ```
   Copy D:\hp\AgentDLL D:\WinPE\i386\system32
   ```

7. Create a directory for storing the HP storage driver:

   ```
   mkdir D:\WinPE\i386\System32\hpStorage1
   ```

8. Copy the HP storage driver to the new directory:

   ```
   copy D:\hp\Storage1\*.* D:\WinPE\i386\System32\hpStorage1
   ```

9.   Open the D:\WinPE\i386\System32\Winpeoem.sif file for editing.

10. Update the Winpeoem.sif file to the following:

```
[OemDriverParams]
OemDriverRoot=""
OemDriverDirs=hpStorage1
```

11. If adding multiple storage drivers, repeat steps 7 through 10 for each storage driver. If adding CPQCISSM or MEGAIDE drivers, refer to the "CPQCISSM and MEGAIDE Drivers" section in this chapter.

12. Copy SmartStart Scripting Toolkit utilities to the Windows PE image:

```
mkdir D:\WinPE\SSTK
xcopy /e /i D:\hp\SSTK\Tools D:\WinPE\SSTK
```

13. Build the ISO image from Windows PE:

```
D:\WinPEBuildTools\oscdimg.exe -n -betfsboot.com D:\WinPE
D:\WinPE.ISO
```

Customizing the Windows PE Image Using a Toolkit Script File

To build a customized Windows PE image using the hp_WinPE_customize.cmd script provided in the Toolkit:

1. Build the Windows PE image:

```
D:\WinPEBuildTools\mkimg.cmd F: D:\WinPE /PnP /WMI
```

2. Build Windows PE optional components if necessary:

  a.   Build the components:

```
D:\WinPEBuildTools\Buildoptionalcomponents.vbs /s:F: /d:
D:\WinPEBuildTools\OptionalComponents /ADO /HTA /WSH /X
```

  b.   Copy the optional components into the Windows PE directory:

```
xcopy /e D:\OPKTools\OptionalComponents D:\WinPE
```

  c.   Edit `startnet.cmd` so that `oc.bat` is launched at the end of the boot process:

```
echo oc.bat >> D:\WinPE\i386\system32\startnet.cmd
```

3. Set up the environment variables:

```
set HPDriversDir=D:\hp\SSTK\HP Drivers\Drivers
set WinPEBuildToolsDir=D:\WinPEBuildTools
set WinPEDestDir=D:\WinPE
```

4. Update the Windows PE image with new device drivers by executing the hp_WinPE_customize.cmd script:

```
hp_WinPE_customize.cmd
```

5. Copy SmartStart Scripting Toolkit utilities to the Windows PE image:

```
mkdir D:\WinPE\SSTK
xcopy /e /i D:\hp\SSTK\Tools D:\WinPE\SSTK
```

6. Build the ISO image from Windows PE:

```
D:\WinPEBuildTools\oscdimg.exe -n -betfsboot.com D:\WinPE
D:\WinPE.ISO
```

3

Creating Utility Data Files

Some Toolkit utilities require data files as input arguments to provide detailed settings for the system. You can build data files manually to use with the utilities, but Toolkit utilities are also capable of capturing existing configuration data files from a server that can then be customized to use with other servers.

NOTE: For more information about building data files manually, refer to the *HP SmartStart Scripting Toolkit Linux and Win32 Editions User Guide.*

Capturing Data Files with Windows PE

SmartStart Scripting Toolkit Win32 Edition utilities are only supported in a Win32 environment. They can run in either Windows or Windows PE. To use the Toolkit utilities in a Windows PE environment, Windows PE must be built with the latest HP drivers to ensure the utilities can access the hardware. For more information, refer to Chapter 2, "Microsoft Windows Preinstallation Environment."

Data files can be captured and stored in either of the following ways:

- Using physical media

- Through a network

NOTE: For more information about using the Toolkit utilities, refer to the *HP SmartStart Scripting Toolkit Linux and Win32 Editions User Guide.*

Capturing and Storing Data Files Using Physical Media

HP SmartStart Scripting Toolkit utilities and scripts can be stored on physical media, such as a diskette or a driverless USB storage device.

NOTE: Windows PE supports USB natively, but USB devices must be connected to the system before Windows PE boots to be recognized. Be sure the proper device driver is loaded if you are using a non-driverless USB device for storage.

After Windows PE boots, you can execute the utilities to capture data files and store them on the USB storage device. These data files can then be modified as needed, and used for deploying configurations on other systems.

Capturing and Storing Data Files Through a Network

For Windows PE to support network access, the driver for the networking device must be on the Windows PE image. To update drivers to the Windows PE image, refer to Chapter 2, "Microsoft Windows Preinstallation Environment."

NOTE: Windows PE does not support access to files or folders on Windows PE servers from another location on the network.

After Windows PE boots, you can access files and folders stored on other servers on the network. Map the proper drives to the file server where the utilities are stored, and execute the utilities to capture data files. These data files can then be modified as needed and used for deploying configurations on other systems.

4

Configuring HP ProLiant Systems with Windows PE

This chapter describes how to perform an unattended configuration using a batch file. It also describes three ways to use Windows PE to configure HP ProLiant systems using captured data files:

- CD-based Windows PE deployment

- Preboot eXecution Environment (PXE)-based Windows PE deployment

- Network-based Windows PE deployment

NOTE: You can also perform an attended configuration. For more information about using the Toolkit utilities in an attended configuration, refer to the *HP SmartStart Scripting Toolkit Linux and Win32 Editions User Guide.*

Unattended Configuration using Startnet.cmd

In an unattended configuration, a batch file is launched automatically at the end of the Windows PE boot process to perform the configuration tasks. There are several ways to start a batch file for custom routines, but the simplest is by using startnet.cmd.

NOTE: For more information about the `startnet.cmd` command and other methods of starting a batch file, refer to the OPK documentation.

Startnet.cmd, located in the *WinPE_Directory\Platform*\system32 directory, is a command batch file that Windows PE automatically loads at the end of the boot process.

The content of startnet.cmd varies depending on usage. In the following example, startnet.cmd is used to map drives to a network file server and execute files located on a remote server.

Sample contents for startnet.cmd:

```
factory –winpe
SystemDrive\util\config.cmd
net use f: \\fileserver\share
f:\generic.cmd
a:\custom.cmd
```

NOTE: For more information about the `net use` command, enter `net help use` at the command prompt.

Because startnet.cmd is stored in the *WinPE_Directory\Platform*\system32 directory, command files listed in startnet.cmd are relative to the directory. Use the default system variables *SystemDrive* and *SystemRoot* to specify paths to the utilities. Be sure that the contents of the file are accurate and tested before using it widely.

CD-Based Windows PE Deployment

This section assumes all scripts, utilities, and data files are stored on the Windows PE CD.

NOTE: It is also possible to store scripts, data files, and utilities on physical media such as a diskette or USB storage device.

NOTE: For information about building a customized Windows PE image, refer to chapter 2, "Microsoft Windows Preinstallation Environment."

The Windows PE CD image must contain the utilities, data files, latest device drivers from HP, and at least one customized batch file to initiate the system configuration.

To build a Windows PE CD image that automatically configures an HP ProLiant system:

1. Build the Windows PE folder structure as described in chapter 2, "Microsoft Windows Preinstallation Environment."

2. Modify *WinPE_Directory\Platform*\system32\startnet.cmd to include the following:

   ```
   factory –winpe
   custom.cmd
   ```

   NOTE: For more information about using a batch file, refer to "Unattended Configuration using Startnet.cmd" in this chapter.

3. Create the file custom.cmd in *WinPE_Directory\Platform*\system32 directory to perform the following tasks:

   — Configure the system BIOS with the appropriate settings.

   — Configure the storage RAID controller if necessary.

   — Configure the Integrated Lights-Out (iLO) if necessary.

   — Reboot the system if necessary.

4. Build an ISO image using the Windows PE folder.

5. Create a Windows PE CD using the ISO image.

PXE-Based Windows PE Deployment

This section describes how to use a PXE server to boot Windows PE through a network. Check the server specification to determine whether PXE is supported on the target server. Currently, the Microsoft Remote Installation Services (RIS) server is the only PXE server officially supported by Microsoft with Windows PE. Third-party PXE servers have not been validated with Windows PE. Refer to RIS documentation for information about setting up and using RIS.

Using PXE to boot Windows PE eliminates the need to create a physical Windows PE CD. Because the Windows PE image files reside in the RIS server, files can easily be modified if needed. With network access, scripts, utilities, and data files can be stored on the RIS server or on other file servers on the network. For detailed information about using RIS with Windows PE, refer to the OPK documentation.

To build a Windows PE image for use with a PXE server to automatically configure an HP ProLiant system:

1. Build the Windows PE folder structure as described in chapter 2, "Microsoft Windows Preinstallation Environment."

2. Store the command batch file on RIS or on other servers on network. In the following example, the customized command batch file is located on *fileserver**share*.

```
factory –winpe
net use f: \\fileserver\share
f:\custom.cmd
```

NOTE: For more information about using a batch file, refer to "Unattended Configuration using Startnet.cmd" in this chapter.

3. Create the file custom.cmd in *WinPE_Directory**Platform*\system32 directory to perform the following tasks:

— Configure the system BIOS with the appropriate settings.

— Configure the storage RAID controller if necessary.

— Configure the iLO if necessary.

— Reboot the system if necessary.

To use the Windows PE image with RIS, refer to the "Booting Windows PE from Remote Installation Services (RIS) Servers" section in the OPK documentation.

Network-Based Windows PE Deployment

This section assumes that the scripts, data files, and utilities are stored on a file server. Windows PE runs startnet.cmd at the end of the boot process and launches the customized command batch file automatically.

To build a Windows PE image to automatically configure an HP ProLiant system through a network:

1. Build the Windows PE folder structure as described in chapter 2, "Microsoft Windows Preinstallation Environment."

2. Modify *WinPE_Directory**Platform*\system32\startnet.cmd to include the following:

```
factory –winpe
net use f: \\fileserver\share
f:\custom.cmd
```

NOTE: For more information about using a batch file, refer to "Unattended Configuration using Startnet.cmd" in this chapter.

3. Create the file custom.cmd in *WinPE_Directory\Platform*\system32 directory to perform the following tasks:

   — Configure the system BIOS with the appropriate settings.

   — Configure the storage RAID controller if necessary.

   — Configure the iLO if necessary.

   — Reboot the system if necessary.

4. Build an ISO image using the Windows PE folder.

5. Create a Windows PE CD using the ISO image.

Index