ImageUltra Builder
Version 3.0

ThinkVantage

User Guide

ImageUltra Builder
Version 3.0

ThinkVantage

User Guide

> **Note**
>
> Before using this information and the product it supports, be sure to read "Notices," on page 191.

Contents

Preface

This guide has three purposes:

- To help you understand the concepts behind building, deploying, and installing Smart Images using the ImageUltra™ Builder program.
- To guide you through the ImageUltra Builder installation process.
- To supplement the ImageUltra Builder help system, which provides detailed step-by-step procedures for the tasks associated with the ImageUltra Builder program.

Before installing or using the ImageUltra Builder program, make sure that you are familiar with the information in Chapter 1, "ImageUltra Builder capabilities," on page 1 and Chapter 2, "An overview of the image process," on page 11.

This guide is organized as follows:

Chapter 1, "ImageUltra Builder capabilities," on page 1 contains an overview of the concepts and functions of the ImageUltra Builder program.

Chapter 2, "An overview of the image process," on page 11 contains an introduction to the image-building process using the ImageUltra Builder program.

Chapter 3, "Staging-area behavior and logic," on page 33 contains information about the staging areas used for deployment and installation and the logic used by the ImageUltra Builder program to adapt to various conditions on the target computer that affect the staging area selected.

Chapter 4, "Installing the ImageUltra Builder program," on page 47 contains installation instructions for the ImageUltra Builder program, as well as prerequisites, limitations, and considerations that you need to understand before you install the program.

Chapter 5, "Working with the ImageUltra Builder interface," on page 55 introduces you to the various windows and conventions used in the ImageUltra Builder interface.

Chapter 6, "Map construction overview," on page 79 focuses on the concepts behind creating a map tree structure and setting some of the map attributes.

Chapter 7, "Creating maps," on page 117 provides step-by-step instructions for creating base maps and driver maps.

Chapter 8, "Preparing source files for modules," on page 135 provides step-by-step instructions for preparing the source files, creating repository entries, setting attributes, and building the module.

Chapter 9, "Specifying the module-installation order," on page 167 provides information about the various methods used to control the order in which modules are installed.

Chapter 10, "Getting help and support," on page 173 contains instructions about getting help and support for the ImageUltra Builder program.

Chapter 11, "Dealing with exceptions," on page 175 contains helpful information about working with special cases that affect the image building process.

Chapter 12, "Debugging problems," on page 183describes ImageUltra Builder's tools for debugging and best practices to use when debugging specific problems.

"Notices," on page 191 contains legal notices and trademark information.

Who should read this guide

This guide is intended for information technology professionals who do the following:
- Set up and maintain computer images
- Deploy images by CD or across a network

Web page references

This guide provides Web-page addresses for additional information. Web-page addresses are subject to change. If you find that a Web-page address does not work, go to the main Web page for the company being referenced (for example, http://www.lenovo.com) and then search for the appropriate term or phrase (for example, ImageUltra Builder).

New features in ImageUltra Builder

Features added to versions 2.0 and 2.01

The following features were added to ImageUltra Builder versions 2.0 and 2.01
- Redesigned Repository window: The Repository window has been redesigned to help you organize, categorize, and sort large numbers of maps and modules.
- Repository search feature: A repository search feature has been added to help you find specific maps and modules. The search engine enables you to pick predefined values or use free-form text to search for strings or phrases associated with map settings or module attributes. To help facilitate the free-form text search, a comment field has been added to the module attributes General tab and the map settings General tab. An advanced repository-search feature enables you to find maps or modules that share a common characteristic. For example, you can find all modules not used in any map or all maps that use a specific module.
- Export Wizard: An Export Wizard has been added to enable you to archive maps and modules in a folder outside of the repository. The archived maps and modules can be restored to a repository, if needed, through the use of the Import Wizard.
- Import Wizard: An Import Wizard replaces the import process and provides the extended capabilities of importing individual modules and maps to the repository from a variety of sources. When importing maps and modules from a source computer, instead of creating and using import media, you simply install the ImageUltra Builder program on the source computer, connect to the target repository, select **Import** from the Main window menu bar, and then follow the instructions provided by the Import Wizard.
- Deploy Wizard: A Deploy Wizard simplifies the process of creating distribution CDs, and network-deployment CDs.
- Redesigned Map window: The Map window has been redesigned so you can select and manipulate multiple items in a map.

- Variable support: Support for variables in the menu system has been added to allow dynamic flow processing and increased flexibility based on user responses during the image-installation process.
- Enhanced control of the module-installation sequence: You can now control the installation sequence and execution of individual modules.
- Customized behavior for a module in a map: You can now alter the behavior of a common module on a map-by-map basis through a new Customization field.
- Additional menu-customization capabilities: You now have better control over the appearance and behavior of the installation menus. You can select background colors, set margins, do window shadowing, enable and disable the Ctrl+Alt+Del key sequence while menus are being viewed, define text to be used in the status line, specify a key to exit the current menu, define a key for returning to a previous menu, define a keyword that takes the user to a hidden system menu, and more.
- Module password protection: You can password protect modules so that only ImageUltra Builder processes have access to the source files inside of a module.
- Custom service-partition size: In the base map, you can now specify the size that you want the service partition to be after the image has been deployed and installed. This feature is useful to streamline the Rapid Restore PC installation processes. By specifying extra room in the service partition, you can eliminate the time required by the Rapid Restore installation process to rearrange data and resize the service partition to accommodate the backup files.
- Support for the Hidden Protected Area (HPA): Many new ThinkPad® and ThinkCentre™ computers use an HPA for recovery and diagnostic purposes instead of the service partition used by their predecessors. The HPA provides increased security, protection, and recovery capabilities. The ImageUltra Builder 2.0 program supports importing maps, modules, and containers from an HPA and sourcing device-driver modules from an HPA during the installation process when a driver map is not used. The import and deployment support for service partitions also remains.
- Direct-network installation to the C partition: This feature enables you to deploy and install an image from the network directly to the C partition without having to copy the maps and modules to a service partition.

Features added to version 2.2

The following features have been added to ImageUltra Builder version 2.2:
- Enhancements to the Deploy Wizard: You can now use the Deploy Wizard to create distribution discs on CD or DVD media, network-deployment discs on CD or DVD media, and ISO images without third-party CD-burning software. Another new feature eliminates the need to create a network-deployment diskette before you create a network-deployment CD. In addition, you can optionally set a static IP address for the target computer instead of depending on one being assigned by a DHCP server.
- Network-deployment CDs now support multiple network cards.
- Enhanced flexibility in sequencing module installation order: The module Options tab now supports up to 9999 install slots for greater control of the module-installation sequence.
- New module templates: Templates are provided for the new I386 module and the unattend.txt module. These templates are in-built modules that can be modified, imported, or exported. Or if you prefer, you can create your own template modules.

- Enhanced flexibility in maps: You can now create menu items that exclude modules that were added to the installation list by previous menu selections.
- Enhanced debugging capabilities: A log is now created on the target computer during the deployment process that indicates when a module installation process is started and completed as well as the result of the filter expressions used in the menu items.

Features added to version 3.0

ImageUltra Builder 3.0 has implemented significant changes on how base maps are displayed at the target computer during deployment. These changes are a result of ImageUltra Builder 3.0 using the Windows Preinstallation Environment (PE) as its deployment environment instead of the DOS operating system used by previous versions. Under DOS, a base map was displayed to users in full-screen, console-like mode. Now, through Windows PE, base maps are presented to users within an actual window.

The following features were added to ImageUltra Builder version 3.0:

- There are new and updated map settings and property tabs within The ImageUltra Builder 3.0 Graphical User Interface (GUI) menu-driven application, which control the appearance of deployment and installation screens at the target computer.
- Improved Map Menu Program: The System Menu Keyword field has been moved from the Menu Appearance tab to the General tab. Since this only applies to base maps, it is hidden when the settings dialog applies to a driver map.
- The use of IBM® Cloudscape™, a client-server, object-relational database system, instead of the previous file-based database system.
- The use of a new boot loader on the ImageUltra deploy media that allows the install process to continue without requiring the user to remove or deploy media.
- The use of a splash window. The splash window is used to display the contents of a log file. A message-box dialog displays messages to users as it moves around the screen.
- The new Migrate Repository Wizard that takes users through the different stages to migrate from a previous version of an ImageUltra repository.

If you are upgrading from a previous version of ImageUltra you need to know the following:

- Portable Sysprep modules need to be rebuilt in ImageUltra Builder 3.0 format
- Ultra portable base maps need to be redesigned to follow the ImageUltra Builder 3.0 format

Chapter 1. ImageUltra Builder capabilities

The ImageUltra™ Builder program provides new methods of creating, deploying, and installing custom images. This chapter provides information to help you understand the concept of building images and the various types of images that the ImageUltra Builder program supports.

Understanding images

An image consists of several components: an operating system, applications, device drivers, and other related files needed to successfully run a computer. In most cases, information technology (IT) departments within corporations or system integrators create images on source computers, extract the images and store them in a central location. Then, they deploy the images to multiple computers either across a network or from a set of image CDs. This enables an IT department to develop standardized images and control the versions of software and device drivers used throughout the company.

Developing and deploying images in this way ensures that every computer within a company maintains a high level of image consistency. However, because images created by this method are hardware dependent, a significant amount of time is spent creating, testing, and transferring images from multiple source (donor) computers to accommodate various hardware configurations and variations of supported software. In many cases, in order to accommodate specific software requirements, a base image is deployed and specific application programs are added manually on a system-by-system basis after the image has been installed.

As a result, IT departments often have multiple images to manage. Whenever a change is required to accommodate new hardware, an operating-system update, a device-driver update, an additional language, or a new or changed application program, many images must be rebuilt.

The major challenges most IT departments face are as follows:
- Reducing the amount of time spent creating and testing images.
- Finding a way to build and deploy images that provides increased flexibility to meet specific software requirements that might vary from department to department.
- Finding a way to build and deploy images across a wider range of hardware.
- Reducing or minimizing the total number of images that have to be maintained.

The purpose of the ImageUltra Builder program is to provide IT departments with the tools needed to develop, deploy, install, and maintain a variety of images efficiently. Through new tools and techniques, the ImageUltra Builder program enables you to break down images into reusable subcomponents (called *modules*) and use *maps* to construct Smart Images that can be deployed across a broad range of hardware platforms.

Definition of a Smart Image

The ImageUltra Builder program enables you to store a variety of image-related files as reusable *modules* in a repository. Typically, there is a module for each operating system, application, and device driver. These modules are organized by *maps* to meet the needs of specific users, groups, or an entire enterprise. A single *base map* can contain multiple operating systems and applications, as well as variations based on language. A single *driver map* can contain device drivers for a single hardware platform, or through the use of *filters*, device drivers for a variety of hardware platforms.

Delivering the maps and modules to a target computer involves two steps: *deployment* and *installation*. At the beginning of the deployment process, the installer selects which base map and driver map to use. The selected maps and the modules defined within the maps are copied to a *staging area*. The compilation of maps and the modules defined within the maps is called a *Smart Image*.

A Smart Image has the following characteristics:
- It is the source from which a specific image is installed on the target computer.
- It provides a menu system during the installation phase. The person controlling the installation process makes selections from the menu system to define the content of the image to be installed. The menu entries are defined in a *base map*.
- A single Smart Image can contain multiple images or components that make up a variety of images.
- If desired, a Smart Image can reside on a target computer after installation and be used for console-side recovery without any dependencies on an active network connection.
- Smart Images can be developed to provide varying degrees of *portability* across different hardware platforms. Portability is the degee in which you can use a single image for different types of computers.

Staging areas

A staging area is a workspace used during deployment and installation. It is where certain modules defined by a map are unpacked, where ImageUltra Builder installation files are stored, and where installation logs are created. The ImageUltra Builder program can use two types of staging areas:
- **Service partition:** The service partition is a hidden primary partition that is protected against normal, end-user read and write operations. This partition is a standard Lenovo™ factory-installed feature on many Lenovo ThinkPad®, NetVista™, and ThinkCentre® computers. As shipped from the factory, the service partition provides built-in recovery and diagnostic capabilities. When a pre-existing service partition is used in conjunction with the ImageUltra Builder program, the deployment phase deletes the factory-installed recovery modules and replaces them with appropriate modules and maps for the image being deployed. If a service partition does not already exist, the ImageUltra Builder program creates one, if needed, as part of the deployment process. After the deployment is complete, the person controlling the installation process (end user or IT professional) can initiate the installation process from the service partition immediately after deployment or at any point in the future without any dependencies on a network connection.
- **Network-staging folder:** This temporary staging area is for a *direct-network installation* only. A direct-network installation combines the deployment phase and installation phase into a single operation, which in almost all cases results in

a faster overall deployment/installation cycle. Speed is affected by the amount of network traffic and certain map settings.

During the deployment phase, a unique network staging-folder is created on the network for the target computer. The person controlling the direct-network installation (typically an IT professional) first selects the base map and driver map. Then, only those modules required to control the installation of the selected maps are obtained from the repository and unpacked into the network-staging folder. When the required modules are unpacked, the person controlling the installation makes selections from the installation menu system to define the specific image required for the target computer. When the image is defined, installation to the target computer automatically begins. All modules that are required to create the image content are obtained from the repository, unpacked dynamically to the C partition of the target computer, and run through a series of setup programs to create the image on the target computer. When the installation is complete, the network-staging folder is deleted from the network. Depending on the settings in the base map being used, any of the following actions can occur at the end of the installation process:

– Pre-existing recovery files in the service partition can be updated to contain the complete Smart Image, thereby providing console-side recovery for any image contained within the Smart Image

– Pre-existing recovery files in the service partition can be updated to contain the maps and modules required to recreate the installed image only, thereby providing console-side recovery for the installed image only.

– Pre-existing recovery files in the service partition can be deleted, thereby eliminating console-side recovery.

Types of images supported by a Smart Image

A Smart Image can contain three different types of images, each achieving different levels of portability:

- **Ultra-Portable Image:** This type of image contains hardware-independent operating-system modules, application modules, and device-driver modules that are either developed by Lenovo or created through the ImageUltra Builder program.

  Using operating-system modules that you develop through the ImageUltra Builder program, you can create Smart Images that can be used on virtually any personal computer platform, regardless of manufacturer. The only limiting factor is the licensing agreement between you and the Microsoft Corporation related to the Product ID (PID) you assign to the operating-system module.

  Using ImageUltra-developed operating system modules, you can create Smart Images that can be used across the IBM branded product line of personal computers. For a listing of IBM branded HIIT-enabled personal computers, visit the ImageUltra Web site at
  `http://www-3.lenovo.com/pc/support/site.wss/MIGR-44316.html`.

- **Portable-Sysprep Image:** This type of image is created by a third-party image-cloning tool, such as Symantec Norton Ghost or PowerQuest DeployCenter, but modified by ImageUltra Builder tools to enable the installation of additional modules created and managed by the ImageUltra Builder program. The additional modules can be device-driver modules to enhance portability across various hardware platforms or application modules to provide additional flexibility to meet the needs of individual departments.

- **Hardware-Specific Image:** This type of image is created by a third-party image-cloning tool, such as Symantec Norton Ghost or PowerQuest DeployCenter. It is a self-contained image, which *does not* enable the installation

of additional modules. The hardware configuration of the target computer must be the same as the hardware configuration of the source computer used to create the image.

Additional detail about these image types are discussed later in this publication.

The use and importance of Sysprep

If you intend to use Portable-Sysprep Images or Hardware-Specific Images, you must have a thorough knowledge of the Microsoft® Sysprep tool. Unlike Ultra-Portable Images, Portable-Sysprep Images and Hardware-Specific Images are based on images extracted from a source computer that have gone through a Windows setup operation. Therefore, specific user and hardware information must be removed from the source computer using the Sysprep tool before the image can be deployed to other computers.

Which image types are best for you?

The types of images you choose to deploy using the ImageUltra Builder program depend on the mix of PCs used by your company, the investment you have in images created by third-party image-cloning software, and the willingness of your IT staff to learn new methods for creating and modifying images.

For example:
- To maintain existing PCs for which you already have images, you might choose to convert existing Hardware-Specific Images into modules that can be stored and deployed using the ImageUltra Builder program.
- If you are planning to install assorted models of IBM branded PCs only, you might consider developing Ultra-Portable Images using imported ImageUltra-developed operating-system modules that can be used across the IBM branded product line regardless of hardware variations.
- If you plan to install a mix of PCs from various manufacturers and want to minimize the number of images you want to maintain and have the maximum portability across various PC manufacturers' platforms, your best bet is to develop Ultra-Portable Images using operating-system modules that you create using the ImageUltra Builder program.
- If you plan to install a mix of PCs from various manufacturers, and you are not ready to make the jump to developing Ultra-Portable Images, consider using the ImageUltra Builder program in conjunction with third-party image-cloning software to create Portable-Sysprep Images that enable the installation of application or device-driver modules so that the image can be deployed across a wider range of PCs.

Because Smart Images support all three types of images, you can use combinations of images within a single base map.

Each image type has its own characteristics. The following sections provide a brief overview of the function, advantages, and disadvantages of each type of image that can be deployed as part of a Smart Image.

Ultra-Portable Image

The Ultra-Portable Image is the most portable of the three types of supported images. Because an Ultra-Portable Image is deployed in an uninstalled state and uses the full Windows setup for hardware detection during installation, the image

is virtually hardware independent. However, the high portability of an Ultra-Portable Image comes with a cost; this type of image takes more than 50 minutes to install. Nevertheless, the life-cycle of an Ultra-Portable Image is typically longer than Portable-Sysprep Images and Hardware-Specific Images because the image can be used across a wide range of personal computer platforms regardless of the core chip set and other hardware variations.

- When using imported ImageUltra-developed base operating-system modules, you can use the same Ultra-Portable Image across the entire IBM branded product line of personal computers, but you cannot use the same Ultra-Portable Image on non-IBM branded computers.

- When using I386 base operating-system modules that you develop using the ImageUltra Builder program, you can use the same Ultra-Portable Image across multiple manufactures' personal computers. The only limiting factor is the licensing agreement between you and the Microsoft Corporation related to the PID you assign to the operating-system module.

Anatomy of an Ultra-Portable Image

An Ultra-Portable Image consists of several modules that are either developed using the ImageUltra Builder program or imported from the service partition or the Hidden Protected Area (HPA) of a source IBM branded computer.

Note: For more information about the HPA, see "ImageUltra Builder relationships with the Hidden Protected Area (HPA)" on page 38 or visit the Lenovo Web site at:
http://www.lenovo.com/think/support/site.wss/
docview.wss?uid=psg1MIGR-46023

These modules are stored in the ImageUltra Builder repository. One type of module is the *base operating-system module*. The base operating-system module contains all of the files required for a specific operating system. An ImageUltra developed base operating-system module resides in the service partition or HPA of the source IBM branded computer and is part of a *container module* that also holds *add-on operating-system modules* for hot fixes, patches, updates, and service packs. Additional modules, such as *application modules* and *device-driver modules* also are located in the service partition or HPA. Any of the modules contained in the service partition or HPA can be imported into the ImageUltra Builder repository. Additionally, the ImageUltra Builder program enables you to import modules from ThinkCenter and ThinkPad Product Recovery CDs and build your own application modules, device driver modules, and operating-system modules.

An Ultra-Portable Image enables you to use a common base module with various combinations of device-driver and application modules to produce department-unique or user-unique images. Because of the modular construction and reuse of modules, Ultra-Portable Images are much more efficient to store than traditional images.

As you are developing an Ultra-Portable Image, the ImageUltra Builder program also enables you to predefine default user settings through an UNATTEND.TXT file or implement a process that prompts the person installing the image to provide all necessary user settings near the beginning of the installation process. The settings are stored and provided to the Windows setup program when needed. This feature minimizes the amount of time that someone needs to be present at a target computer during the installation process.

Ultra-Portable Images are typically limited to a single-partition installation. During the installation process, any unused disk space is made into a single C partition,

and the Ultra-Portable Image is installed in that partition. However, if you need to create one or more additional partitions, you can create a unique type of operating-system module called a *partitioning module* and set the sequence of the installation so the partitioning module is installed before the base operating-system module. The partitioning module contains all of the information required to create the additional partition and all of the data to be placed in the additional partition. You also use partitioning modules to delete one or more existing partitions on a target computer before the new image is installed.

For more information about preparing modules, see Chapter 8, "Preparing source files for modules," on page 135.

Portability of an Ultra-Portable Image

From a technical perspective, the same base operating-system module and most application modules can be used on virtually any personal computer. The high portability of the Ultra-Portable Image is limited only by the PID assigned to base operating-system module.

- I386 base operating-system modules that you develop through the ImageUltra Builder program are virtually hardware independent and limited only by the licensing agreement between you and the Microsoft Corporation that is associated with the PID you assign. You specify the PID in an associated UNATTEND.TXT module either as the Product ID setting in the Windows 2000 version of the UNATTEND.TXT file or the ProductKey setting the Windows XP version of the UNATTEND.TXT file. Additional information about the UNATTEND.TXT module and UNATTEND.TXT source files is discussed later in this user guide.
- ImageUltra-developed base-operating-system modules that you import from a new IBM branded personal computer can be used on other IBM branded personal computers only, not on non-IBM branded computers.

  For a listing of IBM branded HIIT-enabled personal computers, visit the ImageUltra Web site at
  `http://www.lenovo.com/think/support/site.wss/MIGR-44316.html`.

  Note: Do not install an Ultra-Portable Image that contains an imported, ImageUltra-developed base operating-system module on a non-IBM branded computer. If you attempt to install an Ultra-Portable Image on a non-IBM branded computer, the deployment and installation should complete without error; however, you might be violating terms of the Microsoft Windows End User License Agreement and the license agreement for this product. Also, in the case of Windows XP, an Ultra-Portable Image containing an imported ImageUltra-developed base operating-system module will work for only 30 days after installation on a non-IBM branded computer. After 30 days, Microsoft will require you to register the operating system and computer to activate the image.

An Ultra-Portable Image enables you to use a driver map to control the level of the device drivers deployed at the target computer. However, if you are deploying the image to IBM branded personal computers, you have the option of using the Lenovo-factory-installed device drivers already contained in the service partition or the Hidden Protected Area (HPA) of the target computer. Using the factory-installed device drivers simplifies the image building process because you do not have to concern yourself with driver maps and device-driver modules. However, for all practical purposes, using the factory-installed device drivers limits the portability of the image to IBM branded personal computers because those are the only personal computers that have a service partition or HPA that contains

device drivers. In most cases, it is best to manage your device drivers through a driver map. Not only does using a driver map enhance portability, but it enables you to standardize device drivers, thereby reducing diagnostic time if a problem arises.

An added benefit of using your own driver map is that you can use the Image-Ultra Builder program for recovery purposes if a hard disk fails and requires replacement. Because the driver map deploys the device drivers as part of the Smart Image, no preexisting drivers need to be present at the target computer. Therefore, by using a driver map, the Smart Image can be installed on a new hard disk drive.

Portable-Sysprep Image

Using the supported third-party image-cloning tools (Symantec Norton Ghost or PowerQuest DeployCenter) in conjunction with the ImageUltra Builder program, you can create a Portable-Sysprep Image that can be deployed across a wider range of hardware than a traditional image.

Anatomy of a Portable-Sysprep Image

In contrast to an Ultra-Portable Image, a Portable-Sysprep Image is a "snapshot" of a source computer hard disk contents. The source computer is set up to match the desired configuration of the target computers. However, because Windows setup has been run on the source computer, specific user and hardware information (such as the user ID, passwords, and network settings) is recorded in the source computer registry. The Microsoft Sysprep tool must be run on the source computer to erase this information before third party imaging software is used to extract the image.

By implementing a few minor variations during the development of a traditional image, you can enable the image to use application or device-driver modules to append the image during the deployment and installation process. These variations are discussed in detail in Chapter 8, "Preparing source files for modules," on page 135.

After creating the image using Symantec Norton Ghost or PowerQuest DeployCenter, you use the ImageUltra Builder program to create and build a base operating-system module from the image. The module is automatically stored in the ImageUltra Builder module repository during the module-building process.

After a Portable-Sysprep Image has been deployed to the target computer hard disk, a mini-setup runs during the installation process. Any device drivers that are defined in the driver map are copied to the C drive and made available to the mini-setup. The mini-setup prompts for specific user information and performs a limited hardware query. Although the mini-setup makes allowances for minor variations in the hardware (such as audio, video, and so on), it does not deal effectively with different core chipsets. Because a Portable-Sysprep Image does not have to go through a complete Windows setup, the Windows setup time is typically reduced to 10 minutes or less. The ImageUltra Builder program enables you to predefine default user settings or implement a process that prompts the person installing the image to provide all necessary user settings near the beginning of the installation process. These settings are stored and provided to the mini-setup program when needed.

Upon the successful completion of the mini-setup, any additional applications defined by the base map are installed. The actual setup time for a complete image installation varies depending on the size and number of application modules defined in your base map.

Portability of a Portable-Sysprep Image

Although Portable-Sysprep Images can be used with IBM branded and non-IBM branded computers and provide increased portability by enabling additional device drivers and applications to be installed, there are still hardware limitations. As a general rule, a Portable-Sysprep Image is for use on identical or similar computers, and in many cases, the Portable-Sysprep Image is manufacturer unique.

There are various factors in determining what constitutes a similar computer:

- If you are deploying a Portable-Sysprep Image to a target computer that has the identical system board (motherboard) as the source computer, the image will install and work successfully.
- If you are deploying a Portable-Sysprep Image to a target computer that has a different system board, but the same core chip set as the source computer, there is a high probability that the image will install and work successfully.
- If you are deploying a Portable-Sysprep Image to a target computer that has a different system board and a different core chip set from the source computer, there is a low probability that the image will install and work successfully.

Hardware-Specific Image

The least portable image is the Hardware-Specific Image. Hardware-Specific Images can be developed for, deployed to, and installed on IBM branded and non-IBM branded computers, but each image must be unique to a specific hardware and software configuration.

Anatomy of a Hardware-Specific Image

A Hardware-Specific Image is similar to a Portable-Sysprep Image, but it does not have the ability to append itself with additional modules. It is essentially a "snapshot" of a source computer hard disk after the Microsoft Sysprep program has been run. When deployed to a target computer, the Hardware-Specific Image performs a limited hardware query, but the hardware it can support is limited to the device drivers that were provided by the source computer. A Hardware-Specific Image requires that the source and target computers contain the exact same hardware with little or no variations. It also requires that any variation in applications be handled separately from the image-deployment process and image-installation process.

Portability of a Hardware-Specific Image

A Hardware-Specific Image is for use on identical computers. Hardware variations are limited to plug-in devices and peripherals for which device drivers were provided on the source computer. No variation in software content can be handled by the deployment process. If additional device drivers or application programs are required by the target computer, they must be installed manually or through some other software-deployment mechanism.

Summary of image types

The following table summarizes the characteristics associated with an Ultra-Portable Image:

Table 1. Characteristics of an Ultra-Portable Image

Description	Advantages	Disadvantages
• Completely modular architecture • Unpacked at the target computer before Windows setup is run • Uses a full Windows unattended setup, which uses Plug-n-Play hardware detection • User information can be predefined	• When using imported ImageUltra-developed modules, you have maximum portability across the IBM branded product line of personal computers • When using modules you develop through the ImageUltra Builder program, you have maximum portability across IBM branded and non-IBM branded hardware platforms • Extremely reliable • Application programs and device drivers are added independent of the base operating-system module. Therefore, a common base operating-system module can be used with various combinations of driver modules and application modules to produce many variations of an image. • Longer life-cycle of the image • No hardware dependencies • Easy to maintain	• Initial setup at the target computer can take 50 minutes to install the Windows operating system, plus the time it takes to install all applications

The following table summarizes the characteristics associated with a
Portable-Sysprep Image:

Table 2. Characteristics of a Portable-Sysprep Image

Description	Advantages	Disadvantages
• Semi-modular (supports the installation of separate application modules and driver modules to append the base image) • Setup is run once on a source computer • Sysprep cleans out user information and some hardware-specific information from the registry • Performs a limited hardware query at the target computer (Windows Plug-n-Play function) • User information can be predefined	• Initial setup at the target computer is typically 10 minutes or less for a base image, plus the time it takes to install all applications • Some portability (such as video, audio, network, and so on) • Supports IBM branded and non-IBM branded computers • Base operating system module can include some large core applications (such as office suites), which speeds installation time	• Image is used only for identical or similar hardware, and in most cases is manufacturer unique.

The following table summarizes the characteristics associated with a Hardware-Specific Image:

Table 3. Characteristics of a Hardware-Specific Image

Description	Advantages	Disadvantages
• Setup is run once at a source computer • Applications and drivers are part of the base image • Sysprep can be run to clean out user information	• Initial setup at the target machine is typically 10 minutes or less • Images can be developed for, deployed to, and installed on IBM branded and non-IBM branded computers, but each image must be unique to a specific hardware and software configuration	• One image per hardware platform • Image is for use on identical computers only • Unique image must be created for each set of applications, or applications must be added manually after installation. • No portability

Chapter 2. An overview of the image process

This chapter will help you understand the various processes and components associated with the ImageUltra Builder program and how they relate to each other. For step-by-step instructions on how to complete any of the processes described in this chapter, refer to the ImageUltra Builder help system.

The ImageUltra Builder program stores individual image components as modules in the ImageUltra Builder repository. A module is a packaged, compressed, self-contained unit that often can be combined with other modules in the repository to make one or more images. Each application is a module, each device driver is a module, each operating system is a module, and so on.

Note: A module created from a Hardware-Specific Image cannot be combined with any other modules to enhance its function. A module created from a Portable-Sysprep image already contains device drivers and applications, but can be combined with other application modules and device-driver modules to enhance its function.

Modules fall into the following basic categories:

- **Operating-system modules:** There are three types of operating system modules:
  - Base operating-system modules: These modules contain all of the components that are part of the core operating system. Base-operating system modules can be imported from the service partition or Hidden Protected Area (HPA) of a source IBM branded computer or can be created from any of the following:
    - Hardware-Specific Images
    - Portable-Sysprep Images
    - The I386 folder of a Windows installation CD

      Note: The I386 base operating-system module is for use with Ultra-Portable Images only. Whenever a I386 base operating-system module is added to a base map, it must be accompanied by the ImageUltra Customizations module, the Sysprep module, and the appropriate primary partitioning module (Win2000 Primary Partition, WinXP Home Primary Partition, or WinXP Pro Primary Partition) for the operating system you are installing.

      If you want this I386 base operating-system module to install unattended, you must also create an UNATTEND.TXT module and include it in your base map. For more information about these modules and for step-by-step instructions on preparing source files and creating a module, see Chapter 8, "Preparing source files for modules," on page 135.

    Base operating-system modules from Hardware-Specific Images and Portable-Sysprep Images are typically created by and installed through an imaging tool, such as Symantec Norton Ghost or PowerQuest DeployCenter. Imported ImageUltra-developed base operating-system modules and base operating-system modules developed from the I386 folder (hereafter called *I386 base operating-system modules*) go through a complete Windows installation.

– Add-on operating system modules: These modules include items such as operating-system hot fixes, service packs, patches, and updates that install through a standard setup process, similar to applications.

– Partitioning modules: You use these modules to add one or more partitions beyond the C partition, delete one or more pre-existing partitions, or create a partition that can be used with I386 base operating-system modules. You can have multiple partitioning modules in a base map to accomplish various results.

When you use a partitioning module to create additional partitions with data, the module contains all of the information required to create the additional partition and all of the data to be placed in the additional partition. These types of partitioning modules are typically created using an imaging tool, such as Symantec Norton Ghost or PowerQuest DeployCenter. If you just want to add an empty data partition, you can create a simple batch file and use it as the source file for your partitioning module.

For Hardware-Specific Images and Portable-Sysprep Images, you can set the installation sequence to add or delete partitions before or after the new C partition is installed. However, to add a partition after the C partition is installed, you must have created your C partition to leave enough unused disk space to accommodate the additional partition.

Ultra-Portable Images using imported ImageUltra-developed base operating-system modules are typically limited to a single-partition installation because they are designed to use all available disk space. If you need to create one or more additional partitions, you must create a partitioning module and set the installation sequence so the partitioning module is installed before the base operating-system module.

Ultra-Portable Images using I386 base operating-system modules require that you include the appropriate Primary Partition partitioning module for the operating system you are installing. The Win2000 Primary Partition, WinXP Pro Primary Partition, and WinXP Home Primary Partition modules are prebuilt partitioning modules provided as part of the ImageUltra Builder program.

The deployment and installation of operating-system modules are controlled by *base maps*.

- **Application modules:** Each of these modules contains all of the components associated with a specific application program.
  – If you are building an Ultra-Portable Image or a Portable-Sysprep Image, you can use application modules. The deployment and installation of the applications modules are controlled by *base maps*.
  – If you are building a Hardware-Specific Image, you cannot use application modules in conjuction with that image. All applications must be part of the Hardware-Specific Image.
- **Device-driver modules:** Each of these modules contains all of the components associated with a specific device driver.
  – If you are building an Ultra-Portable Image or a Portable-Sysprep Image, you can use device-driver modules. The deployment and installation of the device-driver modules are controlled by *driver maps*.
  – If you are building a Hardware-Specific Image, you cannot use device-driver modules in conjuction with that image. All device drivers must be part of the Hardware-Specific Image.

During the deployment process, you choose which base map and which driver map to deploy. This enables you to maintain device-driver modules and driver maps independently of the operating-system and application content defined by base maps.

In addition to modules, the ImageUltra Builder program enables you to create *containers* in the repository. A container is a special type of module that enables you to group other modules together under a single identifier. All modules in a container must be of the same type. For example, you might want to group all of the device drivers used for a specific machine type into a device-driver container. Or, you might want to group a Windows 2000 base operating-system module with all of its associated add-on modules into an operating-system container.

If you build an I386 base operating-system module, you might want to group the I386 base operating-system module, the appropriate Primary Partition partitioning module, the ImageUltra Customizations module, the Sysprep module, and the UNATTEND.TXT module into a single operating-system container to ensure all required components are kept together.

The use of containers is optional, but you might find containers helpful when creating maps because you can simply insert the container module into your map instead of inserting each individual module.

All maps and modules are stored in a repository. When the repository is viewed through the ImageUltra Builder interface, the maps and modules are identified by descriptive names. The actual file names are assigned by the ImageUltra Builder program. The following is a list of file extensions used for the files in the repository:

- .CRI: Metadata about the module. This information is used within the ImageUltra Builder interface.
- .DMA: Driver maps
- .BMA: Base maps
- .IMZ: Compressed module source

The five basic steps to develop an image

There are five basic steps to developing an image:

1. Importing pre-existing modules
2. Building a module
3. Creating and modifying base maps and driver maps
4. Deploying Smart Images to target computers
5. Installing images on the target computer

The remainder of this chapter provides details about these five basic steps.

Importing pre-existing maps and modules

The Import Wizard copies pre-existing maps and modules into your repository. You can import maps and modules from any of the following sources:

- A service partition on a computer
- The Hidden Protected Area (HPA) on an IBM branded computer
- A directory on a computer
- A different repository

- A set of *ImageUltra Builder Distribution CDs* or *ThinkCenter and ThinkPad Recovery CDs*
- An ImageUltra export package
- A previous version of an ImageUltra repository

Importing maps and modules from a service partition or HPA

To import modules from a source computer, you must first install the ImageUltra Builder program on the source computer and set up the source computer so it has access to the your central repository. After the ImageUltra Program is installed, open the ImageUltra Builder program, open the Repository window for your central repository, then start and use the Import Wizard to import the desired modules to your central repository.

The Import Wizard provides a categorized view of all application modules, device-driver modules, filter modules, utility modules, base maps, and driver maps that are in the source computer service partition or HPA. It enables you to select a category and import all modules within that category, or select only those modules or maps that you want to import.

In many cases, the operating-system modules are in a container and the entire container must be imported. As part of the import operations, the Import Wizard prompts you to select the folder in the repository where you want the maps and modules to reside. You can use an existing folder in your repository, create a new folder in your repository, or just put the imported modules in the root entry of your repository.

The advantages of importing modules, as opposed to building your own, are as follows:
- ImageUltra-developed modules have been tested by Lenovo and are ready for deployment on IBM branded computers.
- *Base maps* and *driver maps* are automatically created by the ImageUltra Builder Import Wizard based on the modules you select to import. You can use these maps as they are, create new maps based on the maps created by the Import Wizard and modify the maps to meet your particular needs, or discard the maps and create your own maps.
- If base maps and driver maps already exist in the service partition of a source computer, you can import those maps. Maps exist in a service partition only if the source computer contains a Smart Image that was deployed by the ImageUltra Builder program.
- ImageUltra-developed modules contain applications and signed device drivers that conform to Microsoft certification requirements, ensuring device-driver compatibility.
- Importing ImageUltra-developed modules from the service partition of a new IBM branded computer provides a streamlined path for creating Ultra-Portable Images that can be used across the entire product line of IBM branded computers.

Importing maps and modules from repositories, folders, and CDs

In addition to importing from the service partition or HPA (as discussed in the previous section), you also can use the Import Wizard from any ImageUltra

Builder console to import maps and modules from a folder, *ImageUltra Builder Distribution CD*, *ThinkCentre and ThinkPad Product Recovery CD*, or a different repository.

Note: You can order a set of *ThinkCentre and ThinkPad Product Recovery CDs* through your local customer support center team. On many IBM branded personal computers announced after 2003, you can create your own set of *ThinkCentre and ThinkPad Product Recovery CDs* using the Access IBM or ThinkVantage toolset.

- When importing from an *ImageUltra Builder Distribution CD* or an *ThinkCentre and ThinkPad Product Recovery CD*, the Import Wizard provides a categorized view of all the application modules, device-driver modules, filter modules, utility modules, base maps, and driver maps that are available and enables you to select all items in that category or select only those modules or maps that you want to import. In most cases, the operating-system modules are in a container and the entire container must be imported.

- When importing from a folder, the Import Wizard provides a categorized view of all the operating-system modules, application modules, device-driver modules, filter modules, utility modules, base maps, and driver maps that are available and enables you to select all items in that category or select only those modules or maps that you want to import.

- When importing from a different repository, you can choose to use either a categorized view or a view of the entire repository as it is currently structured and select only those maps or modules that you want to import.

Building a module

In addition to importing modules, you can create your own modules for operating systems, applications, and device drivers. However, there are a few limitations:

- Application modules and device-driver modules are for use in Ultra-Portable Images and Portable-Sysprep Images only; they cannot be used in conjunction with Hardware-Specific Images. Hardware-Specific Images must have all required applications and device drivers installed on the source computer before the image is created.

- Base operating-system modules can be created from Hardware-Specific Images, Portable-Sysprep Images, or from the I386 folder of a Windows installation CD. The base operating-system module created from an I386 folder (called an *I386 base operating-system module*) is for use with Ultra-Portable images only and must be used in conjunction with the appropriate Primary Partition partitioning module (*Win 2000 Primary Partition*, *WinXP Home Primary Partition*, or *WinXP Pro Primary Partition*), ImageUltra Customizations module, and the UNATTEND.TXT module.

Optionally, you can import ImageUltra-developed modules from the service partition of a new IBM branded personal computer. However, imported ImageUltra-developed base operating system modules are limited to deployment on IBM branded personal computers only.

Building a driver module, application module, or add-on operating-system module

Preparing files for a device-driver module is different from preparing files for an application module or an add-on operating-system module:

- **Application files and add-on operating-system modules:** In general, any application or add-on operating-system component that you intend to use for a

module must have a silent-install capability for unattended installation. Before you build a module, prepare the application or add-on operating-system component for unattended installation so that the installation process does not require any user interaction. In most cases, the Microsoft Software Installer (MSI) and InstallShield programs allow for these types of automated installations.

The ImageUltra Builder program does make allowances for applications and add-on operating-system components that do not allow for an unattended installation. These types of application programs and add-on operating-system components can be deployed as modules, but cannot be automatically installed. Instead, you can have the ImageUltra Builder program copy the installable files to the target computer and put an icon on the desktop that enables the user to launch the setup program and install the program manually.

- **Device-driver files:** When building a device-driver module, you must use the standard Microsoft Plug-n-Play.INF installation method, which is usually unattended in nature. The device driver itself must be certified by the Microsoft Windows Hardware Qualification Lab (WHQL) and signed. Failure to use signed, WHQL-certified device drivers will result in interruptions during the installation process because of Windows-generated information messages, which require user interaction to bypass.

After the application, add-on operating-system component, or device driver has been prepared, use the New Module Wizard to create a module entry in the repository and to set the attributes associated with the module. Attributes vary depending on whether the module is a device-driver module, an application module, or an add-on operating-system module. In all cases you must define a name that will identify the module in the repository and the location where the prepared files are currently stored. When you have finished defining the attributes, you can select the entry from the repository and use the build tool to build the module.

Note: The ImageUltra Builder program does *not* automatically scan files for viruses. To help ensure a virus-free environment, use a virus-checking program on the computer used to build the modules.

For step-by-step instructions on preparing source files and creating a module, see Chapter 8, "Preparing source files for modules," on page 135 or refer to the ImageUltra Builder help system.

Creating a base operating-system module

If you choose to create your own base operating-system modules instead of importing an ImageUltra-developed base operating-system module, you can do any of the following:

- Build a module from an existing Hardware-Specific Image (an industry-standard image created by Symantec Norton Ghost or PowerQuest DeployCenter)
- Create a Portable-Sysprep Image and build a module from it. When you create a base operating-system module from a Portable-Sysprep Image, the module can use application modules and device-driver modules to enhance its portability across a wider range of hardware.
- Create an I386 base operating-system module for use in Ultra-Portable Images.

Creating a base operating-system module from a Hardware-Specific Image

A Hardware-Specific Image is a self-contained entity (operating system, applications, and device drivers) that is prepared using the Microsoft Sysprep tool

and created using either Symantec Norton Ghost or PowerQuest DeployCenter. It has limited portability because the source computer and target computer must use identical hardware with very few variations. Many large companies are using Hardware-Specific Images that can be converted to modules and deployed using the ImageUltra Builder program.

Building an operating-system module from an existing Hardware-Specific Image is simple. You use the New Module Wizard to create an operating-system module entry in the repository and to set the attributes associated with the module. When you have finished defining the attributes, you can select the entry from the repository and use the build tool to build the module.

For step-by-step instructions on preparing source files for a Hardware-Specific Image and creating a module, see Chapter 8, "Preparing source files for modules," on page 135 or refer to the ImageUltra Builder help system.

Creating a base operating-system module from a Portable-Sysprep Image

Creating a module from a Portable-Sysprep Image is nearly identical to creating a module from a Hardware-Specific Image. The only difference is in the way you create the Portable-Sysprep Image on the source computer.

By altering how the image is created, you can enable the image to append other application programs and device drivers during the deployment and installation process. The following are the alterations required to create a Portable-Sysprep image.

Run the ImageUltra Customization Program (provided by the ImageUltra Builder program) on the source computer before you run the Sysprep program. The ImageUltra Customization Program provides the hooks required for Portable Sysprep Images to make use of applications and device drivers defined in maps.

Most other aspects of developing a Portable-Sysprep Image are the same as developing a traditional Hardware-Specific Image including running the Sysprep tool and creating the image files using either Symantec Norton Ghost or PowerQuest DeployCenter. Once you have created the image files, you use the New Module Wizard to create an operating-system module entry in the repository to set the attributes associated with the module. When you have finished defining the attributes, you can select the entry from the repository and build the module. For step-by-step instructions on building a Portable-Sysprep Image, see "Creating a Portable-Sysprep Image" on page 142.

Creating an I386 base operating system module

An I386 base operating-system module is the central element of an Ultra-Portable Image; however, when an I386 base operating-system is added to a base map, it must be accompanied by the appropriate Primary Partition partitioning module (Win2000 Primary Partition, WinXP Home Primary Partition, or Win XP Pro Primary Partition), the ImageUltra Customizations module, the Sysprep module, and optionally the UNATTEND.TXT module. This section provides an overview on how to create and implement these modules.

The I386 base operating-system module contains all of the files and subfolders contained in the I386 folder of a Windows installation CD. These files contain the content and installation procedures associated with installing the operation system. You must create an I386 base operating-system module for each operating system

you want to deploy as an Ultra-Portable Image. The only operating systems that can be deployed as Ultra-Portable Images are:
- Windows XP Professional
- Windows XP Home
- Windows 2000

To build an I386 base operating-system module, you use the New Module Wizard to create an operating-system module entry in the repository. When you have finished defining the attributes, you can select the entry from the repository and use the build tool to create the module. For more information on creating an I386 base operating system module, see "Preparing source files for an I386 base operating-system module" on page 135.

The Primary Partition partitioning module is responsible for preparing the target hard disk to accept a Windows installation. It creates the partition, formats the partition, and establishes the partition as a bootable partition. Three Primary Partition partitioning modules are provided as part of the ImageUltra Builder program:
- The *Win2000 Primary Partition* partitioning module
- The *WinXP Home Primary Partition* partitioning module
- The *WinXP Pro Primary Partition* partitioning module

All of these are ready-to-use operating-system partitioning modules. You must match the appropriate Primary Partition module with the I386 base operating-system module.

The ImageUltra Customizations module contains the ImageUltra Customization program, which provides the hooks required for the Ultra-Portable Images to make use of applications and device drivers defined in maps. The ImageUltra Customizations module is a ready-to-use operating-system module provided as part of the ImageUltra Builder program, and is for use with I386 base operating-system modules only. The ImageUltra Customizations module provided as part of the ImageUltra Builder program is different from similarly named modules that can be imported from source IBM branded computers in that it does not contain the Sysprep function.

Note: Do not attempt to use the ImageUltra Customizations module with Portable-Sysprep Images or an Ultra-Portable Image developed from imported ImageUltra-developed base operating-system modules. Also, some imported ImageUltra-developed application modules might have a dependency on the Sysprep function and might not install correctly when used in images containing an I386 base operating-system module and an ImageUltra Customizations module. If an imported ImageUltra-developed application module does not install correctly, recreate the application module using the ImageUltra Builder program.

The UNATTEND.TXT module is required if you want the image to be installed unattended. This module enables you to replace the default UNATTEND.TXT file contained in the I386 folder with one you create. This file enables you to predefine values needed during the installation process, such as the product key, user name, and company name. In some cases, you might want to create multiple UNATTEND.TXT modules and let the person who will deploy the image select the appropriate UNATTEND.TXT module.

To create this module, you must first create an UNATTEND.TXT file that contains the values you want to use. Once you have created the UNATTEND.TXT file, you use the New Module Wizard to create an operating-system module entry in the repository and to set the attributes associated with the module. When you have finished defining the attributes, you can select the entry from the repository and build the module.

Creating a partitioning module

A partitioning module is a special type of operating-system module that is used for several functions. You can use a partitioning module to delete one or more partitions on a target computer before installing an image and to create additional partitions, beyond the C partition. You can have multiple partitioning modules in a base map to achieve various results. Because an Ultra-Portable Image installation is typically limited to a single-partition, you must create a partitioning module if you intend to install an Ultra-Portable Image and need to have more than one user partition on the target computer hard disk.

If you are installing a multiple-partition Portable-Sysprep Image or Hardware-Specific Image, the use of a partitioning module for the creation of the additional partitions is optional because the Symantec Norton Ghost and PowerQuest DeployCenter tools enable you to capture multiple-partition images from a source computer.

The advantage of using a partitioning module to create additional partitions is that you can use your base map to select the appropriate data partition for the image being installed. For example, if you are installing an image for an accounting department, you can have a separate data partition that contains data specifically for the accounting department.

To create a partitioning module that contains data, you must use either Symantec Norton Ghost or PowerQuest DeployCenter to capture the partition from a source computer and create the installation script for that partition. The installation script defines the additional partition size and contents. Or, if you just want to create an empty data partition, you can write a simple batch file and use it as the source file for your partitioning module. In either case, you use the base map to define which partitioning module to make available for installation, the conditions under which it is to be installed, and module-installation order. You must install partitioning modules *before* Ultra-Portable base operating-system modules. If you are using an I386 base-operating system module for your Ultra-Portable Image, you must install optional partitioning modules *before* the required Primary Partition partitioning module.

Another use of a partitioning module is to preserve one or more preexisting data partitions on a target computer. The Image Ultra Builder program provides one such partitioning module, named *ImageUltra 3.0 - Drive Prep - Delete C Partition Only*. When this partitioning module is correctly implemented in a map, all partitions on the target computer are preserved, except for the C partition. The new image is then installed in the space previously used by the old C partition. Several other partitioning modules are also provided as part of the ImageUltra Builder program. See "Using a partitioning module in a base map" on page 96 for more information.

Creating or modifying base maps and driver maps

To understand the importance and use of base maps and driver maps, you need to have some insight into the deployment and installation processes.

If you are performing a standard network deployment using a service partition as a staging area, the following takes place:

- The deployment process begins by starting the target computer from an *ImageUltra Builder Network Deployment CD* or *ImageUltra Builder Network Deployment CD*. During the deployment process, the person at the target computer is prompted to select a base map and a driver map from the repository. The modules that are defined in the selected base map and driver map are deployed from the repository to the service partition of the target computer.

- During the installation process, the person at the target computer is presented a set of menus. The menu selections define the content of the image that will be installed on the target computer. The menu items and menu hierarchy are defined by menu-item entries in the *base map*. The physical appearance and behavior of the menus are controlled by the *properties* assigned to the various menu items. Based on the menu selections, the appropriate modules are obtained from the service partition and unpacked on the C partition. Then, the setup process begins.

If you are performing a direct-network installation, the deployment and installation are combined into a single operation as follows:

- The direct-network installation begins by starting the target computer from an *ImageUltra Builder Network Deployment CD* or *ImageUltra Builder Network Deployment CD*.

- The person at the target computer is prompted to select a base map and a driver map from the repository.

- When the appropriate maps are selected, a network-staging folder is created on the network for the target computer.

- Only those modules that are required to control the installation are copied to the network-staging folder and unpacked.

- The person at the target computer is then presented with a set of menus. The menu selections define the content of the image that will be installed on the target computer. The menu items and menu hierarchy are defined by menu-item entries in the *base map*. The physical appearance and behavior of the menus are controlled by the *properties* assigned to the various menu items.

- Based on the menu selections, the appropriate modules are copied directly from the repository to the C partition and unpacked. (Optionally, for console-side recovery purposes, the entire Smart Image or the just the modules required for the image being installed can also be copied to the service partition).

- Then, the setup process begins using the unpacked modules on the C partition.

If you are performing an installation from an *ImageUltra Builder Distribution CD*, the following takes place:

- The deployment process begins by starting the target computer from the *ImageUltra Builder Distribution CD*. During the deployment process, the modules that are defined in the base map and driver map on the CD are automatically deployed from the CD to the service partition of the target computer.

Note: The service partition is always used as the staging area when doing an installation from an *ImageUltra Builder Distribution CD.*

- During the installation process, the person at the target computer is presented with a set of menus. The menu selections define the content of the image that will be installed on the target computer. The menu items and menu hierarchy are defined by menu-item entries in the *base map.* The physical appearance and behavior of the menus are controlled by the *properties* assigned to the various menu items. Based on the menu selections, the appropriate modules are copied from the service partition to the C partition and unpacked. Then, the setup process begins.

There are three major parts to every map:

- The tree structure: A map tree structure is very similar to the tree structure found in Windows Explorer; however, instead of containing files and folders, a map tree structure contains menu items and modules, and possibly containers.
- Settings: Each map has a Settings window that enables you to control how a map behaves. The Settings window for a base map also enables you to control the physical appearance of the installation menus.
- Properties: The root entry of the tree structure and each menu item in the tree structure has a set of properties. The properties associated with the root entry and each menu item in a base map enable you to control the flow, the text of the menu item, and the physical appearance and behavior of submenu items. These properties also enable you to associate filters and utilities with menu items and set conditions that control certain behavior on a menu-item by menu-item basis. The menu-item properties for a driver map enable you to associate filters with each menu item and set conditions that affect how the filters are used.

Base maps and driver maps have different purposes and use the tree structure, menu items, settings, and properties differently. Therefore, an individual overview of each is provided in this section. For more detailed conceptual information about maps, see Chapter 6, "Map construction overview," on page 79. For step-by-step information on creating and working with maps, see the ImageUltra Builder help system.

Base maps

The tree structure in a base map enables you to define multi-tiered, multi-branch structures containing a series of menu items that control the flow and content of the menus displayed at the target computer during the installation process. Modules are inserted in the map under menu items in strategic locations. As each menu item is selected at the target computer, modules associated with that menu item are collected. When the final menu item is selected from the last menu in the sequence, all of the accumulated modules are installed.

You use the New Map Wizard to create a base-map entry in the repository. Then, you populate the tree structure with menu items to establish the flow and content of your menu system. Finally, you insert modules under menu items in strategic locations to meet your particular needs. Optionally, you can change the overall appearance of the menu system by modifying the properties associated with the root entry of the tree structure. You can change colors, add window shadowing, control navigation aids, and so on. For detailed information about constructing base maps, see Chapter 6, "Map construction overview," on page 79 and Chapter 7, "Creating maps," on page 117.

Driver maps

Driver maps are for use with Ultra-Portable Images and Portable-Sysprep Images only; driver maps are *not* used by Hardware-Specific Images. If you select a driver map in conjunction with a base map that contains a Hardware-Specific Image, the driver map is ignored during the installation of the Hardware-Specific Image.

The tree structure in a driver map enables you to define a multi-branch structure containing device-driver modules. The use of menu items is optional. Menu items in a driver map serve two purposes only:

- They enable a developer to annotate or label groups of device-driver modules.
- They enable a developer to assign *filters* to one or more device-driver modules. Using filters is optional, but they might be useful if you create a driver map that contains device-driver modules for multiple computer types. The filters provided with the ImageUltra Builder program enable you to key off of the machine type, machine type and model, hardware platform (desktop or mobile), and whether or not a computer is an IBM branded computer.

Menu items in driver maps have no effect on the menus displayed at the target computer.

Use the New Map Wizard to create a driver-map entry in the repository. Then populate the tree structure with menu items, if desired. Finally, insert modules in your tree structure and optionally assign filters to menu items. For detailed information about constructing driver maps, see Chapter 6, "Map construction overview," on page 79.

The use of driver maps is optional if the target computer is an IBM branded personal computer. If you do not select a driver map during the deployment of an Ultra-Portable Image or a Portable-Sysprep Image, the installation process will use the device-driver modules that are present in the target computer service partition or HPA.

It is important to note that when a driver map is used with an IBM branded personal computer, the driver map does not append the factory-installed device-driver modules; it deletes all factory-installed device-driver modules and replaces them with the device-driver modules defined in the map. Therefore:

- If you use a driver map with an Ultra-Portable Image, the driver map must contain *all* device drivers required for the target computer.
- If you use a driver map with a Portable-Sysprep Image, the driver map must contain all device drivers required to supplement those already contained in the base operating-system module.

If you develop an Ultra-Portable Image for non-IBM branded computers, you must include one or more driver maps to support all of the hardware configurations on which this image will be installed.

Defining the deployment and installation method

As you create your base map, the New Map Wizard enables you to define the type of network installation to use. You can choose:

- Direct-network installation: This method uses the network-staging folder to control the installation. Modules are deployed and unpacked directly to the C partition of the target computer.

- Network deployment with local installation: This method uses the service partition as a staging area for the complete Smart Image.

You can also elect not to predefine an installation method and instead, prompt the person at the target computer to select an installation method at the time of deployment.

After you have used the New Map Wizard to create the map, you can change this setting through the map Settings window. See "Specifying base-map settings" on page 121 or the ImageUltra Builder help system for details.

Defining service-partition behavior

As you create your base map, the New Map Wizard enables you to define how the ImageUltra Builder program uses the service partition on the target computer after an image has been installed. You have three options:

- Delete none:
  - For a standard installation (network-deployment with local installation or installation from CD), the service-partition size and contents remain the same as when the Smart Image was deployed.
  - For a direct-network installation, based on the map settings, a service partition is created if one does not already exist. If a service partition does already exist, it is resized if necessary, and the files in the service partition are replaced with ImageUltra maps and modules.

  The end result for either type of installation is that the service partition contains the complete Smart Image.

- Delete some:
  - For a standard installation (network-deployment with local installation or installation from CD), all modules not used for this specific computer image are deleted. The service partition is resized to accommodate only those modules that remain. The C partition is resized to include any free space that was created when the service partition was resized.
  - For a direct-network installation, a service partition is created if one does not already exist. If a service partition already exists, it is resized if necessary, and any preexisting maps and modules in the service partition are replaced with the ImageUltra maps and modules required to recreate the installed image; preexisting utility files are not affected.

  The end result for either type of installation is that the service partition contains only those files required to recreate the image, and the service partition will not be any larger than absolutely necessary to house those files. However, the installation time is typically longer because of the time required to resize partitions and move data.

- Delete all:
  - For a standard installation (network deployment with local installation), the service partition, which was used as a staging area for deployment, is deleted. The C partition is resized to recover any free space that was created when the service partition was deleted. The end result is no console-side recovery at the target computer.
  - For a direct-network installation, the behavior depends on whether or not a service partition existed on the target computer *before* the direct-network installation was initiated. If a service partition already existed on the target computer before the direct-network installation took place, the service partition is deleted and the C partition is resized to recover any free space that was created when the service partition was deleted. If a service partition

did not exist before the installation, none was created, and no action is required; and, in most cases the time required to do the overall deployment and installation is substantially less than any other method. In either case, the end result is no console-side recovery at the target computer.

You can change the setting anytime during the map-development process through the map Settings window. See "Controlling the behavior of the service partition" on page 96 or the ImageUltra help system for details.

Note: If you intend to install the IBM Rescue and Recovery™ with Rapid Restore program as an application in your image, be aware that these programs might use the service partition to store backup images and other utilities. Using either the **Delete all** or **Delete some** setting will prevent these programs from operating if the program is installed as part of the normal silent-install installation process. If you need to use either the **Delete all** or **Delete some** setting, install the IBM Rescue and Recovery with Rapid Restore program *after* the installation process is complete. Also, to avoid multiple partition-resizing operations, you can specify an oversized service partition in the partition tab of the base-map Map Settings window to accommodate the space requirements for these programs.

The IBM Rescue and Recovery with Rapid Restore program is not part of the ImageUltra Builder program. However, many ThinkPad, ThinkCentre, and NetVista computers provided one of these programs as part of the factory-installed software. These programs have been discontinued and replaced by the latest version of the Rescue and Recovery program which can be downloaded from the Lenovo Web site at: http://www.lenovo.com/think/support/site.wss/MIGR-4Q2QAK.html

Enabling the network-sync function

The network-sync function is used to check the repository for updated maps. The check will occur twice if the target computer is connected to the network:
- At the beginning of deployment: If a down-level map is selected, the network-sync function will locate the latest map and use it in place of the down-level map.
- At the beginning of installation: This check is most valuable when the installation process takes place at a later date than the deployment process. In this case, maps that have been updated or replaced since the original deployment process took place are used. Depending on how much the map content has changed since the original deployment, in some cases, this check might result in the deployment of a complete, new Smart Image.

In order for a network-sync operation to locate a replacement map, two conditions must be met:
- The network-sync function must be enabled in the base map.
- The network-sync settings table (accessible through the ImageUltra Builder Tools menu) must contain an entry defining the deployed map and the replacement map.

As you create your base map, the New Map Wizard enables you to designate whether you want to enable the network-sync function. You can change the network-sync settings anytime during the map development process through the map Settings window. See "Controlling the network-sync function" on page 95 or the ImageUltra Builder help system for details.

Defining users and administrators

The new ImageUltra Administrator tool guides administrators through the process of managing servers and users, and users and their access to repositories. The Administrator tool secures repositories based on predetermined access levels set by the administrator.

To help administrators manage and work with large numbers of users, the ImageUltra Administrator provides tasks that organize and manage users limitations and accessibility privileges to repositories and their contents. The Administrator tool provides the following options to manage users and their access to repositories:

- Administer users
- Change a user
- Create a new user
- Control users' access to repositories
- Administer repositories
- Delete the repository
- Unlock a map or module within the repository

To start the tool, click Tools in the Main window menu bar; then, click Administration. Follow the instructions on the screen.

Defining user information

The User Information tab in base maps enables you to define default values for user-specific settings, or prompt for user-specific settings near the beginning of the installation process, or both. You can define the user information settings and prompts anytime during the base-map development. For detailed information about using this feature, see "Defining user information in a base map" on page 93.

If you are creating an Ultra-Portable Image using an I386 base operating-system module, you *must* create an UNATTEND.TXT file to define certain user information and control the operating-system installation. The UNATTEND.TXT file is an answer file that resides in the I386 folder and contains all of the information required by Windows setup. Although you can use the User Information tab in the base maps to provide some of this information, it does not provide all of the fields required for this type of installation. For information about creating or modifying an UNATTEND.TXT file, use the following resources:

- The Microsoft Web site at http://www.microsoft.com. Search for the string UNATTEND.TXT.
- Your Windows 2000 installation CD. Locate the UNATTEND.DOC file.
- Your Windows XP installation CD. Open the DEPLOY.CHM file located in the DEPLOY.CAB file.

After you create your UNATTEND.TXT file, you must create an UNATTEND.TXT module and include it in your base map. During the image-installation process, the UNATTEND.TXT module replaces the UNATTEND.TXT file in the I386 folder with the UNATTEND.TXT file you created. This feature enables you to create multiple UNATTEND.TXT files that can be associated with a common I386 base operating-system module.

Deploying Smart Images to target computers

Important: Before deploying a Smart Image to a real work environment, test the Smart Image to ensure that it works as expected.

You can deploy Smart Images to target computers using either of two methods:

- Direct deployment from *Distribution discs*: This method copies all modules defined in the selected base map and driver map to a set of recordable CD or DVD discs.

  The first CD of the set is bootable. When started in the target computer, the deployment program on the disc copies the modules from the discs the service partition and prepares the target computer for the installation process. If a service partition does not exist at the start of the deployment process, one is created.

  You use the ImageUltra Builder Deploy Wizard to create a set of stand-alone *Distribution discs*. Based on your responses, the Deploy Wizard prompts you through the steps required to create the discs. Typical steps include the following:

  - Creating a *Network Deployment CD*
  - Selecting the base map and driver map to be used
  - Selecting the location of your recordable CD or DVD drive where the CD files are to be stored

  After all required questions have been answered, the Deploy Wizard uses its integrated disc-recording software to create your *Deployment disc* set.

  You can also use the Deploy Wizard to do the following:

  - Create ISO images that can be stored or made available to other locations electronically.
  - Record discs from previously stored ISO images
  - Copy raw files to folders so discs can be created using the third-party disc-burning software of your choice

    Note: Because disc 1 must be bootable, and uses specific files from the *Network Deployment CD*, make sure you follow the instructions provided by the Deploy Wizard and the ImageUltra Builder help system for creating *Distribution discs* from files in folders.

- Standard network deployment (network-deployment with local installation): You use the ImageUltra Builder Deploy Wizard to create either an *ImageUltra Builder Network Deployment CD* or an *ImageUltra Builder Network Deployment disc* on recordable CD or DVD media. Based on your responses, the Deploy Wizard prompts you through the steps required for creating the CD or DVD. Typical information required to create the CD or DVD include the following:

  - Path to the repository
  - IP address (optional)
  - User name
  - Password (if required)
  - Language and keyboard layout
  - Keywords (allows only those maps with matching keywords to be selected from the repository)
  - Location of the drive where the media is to be recorded

Note: Providing User name and password information on the deployment media (CD, or DVD) is optional. If you do not provide a user name and password when you create the deployment media, the person controlling the deployment process will be prompted for this information (if it is required to access the network) when the deployment media is started at the target computer. If you do provide a user name and password when you create the deployment media, it is stored in an encrypted format on the deployment media.

If you choose to make a CD or DVD, in most cases you will use the disc-recording software that is integrated into the ImageUltra Deploy Wizard and the default network device drivers provided by the ImageUltra Builder program to create the CD or DVD. However, in some cases, you might have a need to create a *customized Network Deployment disc* with special network device drivers. The ImageULtra Deploy Wizard provides accommodations to make a *customized Network Deployment disc*. You can use the Deploy Wizard to create a *Network Deployment CD* first and modify the device drivers on it, or make your own boot CD with customized drivers. Then, you use your disc-recording software of choice to create the CD or DVD using the custom Network Deployment CD as source.

Note: Because this CD or DVD is bootable, make sure you follow the instructions in the Deploy Wizard or ImageUltra Builder help system for creating a *custom Network Deployment disc*.

To add device drivers to a deployment CD, do the following:

1. Determine where the device driver files need to go on the CD. That is, the .INI file need to go in the `\i386\INF` directory, and the device driver files need to go in the `\i386\system32\drivers` directory.
2. Create a `Drivers` directory in the repository share directory.
3. In order to separate the NIC software, you need to create a folder for each NIC device driver. For example, `repositoryfilesshare\drivers\Broadcom`, `repositoryfilesshare\drivers\Intel`.
4. You then need to create a directory within the NIC folder and then place the files in it.

Note: Every time you create a Network Deployment CD, the process will automatically retrieve these files and place them on the deployment CD.

Depending on the type of removable-media drive available at the target computer, you use either the CD or DVD to establish a network connection between the target computer and the ImageUltra module repository. After the connection is established, you select which base map and driver map to use. All of the modules defined in the map are downloaded to the service partition or HPA of the target computer, and the target computer is prepared for the installation process. If a service partition or HPA does not exist at the start of the deployment process, one is created.

The advantages of CD or DVD deployment are speed and the flexibility of not having to be dependent on a network connection. The advantage of network deployment is the assurance that the latest maps and modules will be installed if the network-sync function is enabled.

With the exception of a direct-network installation, the division between the deployment and installation is a single boot cycle. In many cases, deployment is done in one location and installation takes place in another location, typically at

the end-user's work area. When deployment is complete, you can shut down the target computer and prepare the computer for delivery to the end-user's work location.

Note: Some large enterprises that use drive-duplication technology for mass production might want to replicate the drive at the end of the deployment process, prior to installation. In order for this drive-duplication process to work successfully, you must install the IBM Boot Manager on each drive *after* the drive has been duplicated. You can get the IBM Boot Manager installation tool (BMGRINST.BAT) from the TOOLS\BMGR folder of your repository. You must run BMGRINST.BAT during an WinPE session.

Installing images on the target computer after deployment

In a standard network installation (network-deployment with local installation) or an installation from an *ImageUltra Distribution CD*, the first boot cycle following the end of deployment starts the installation process. The following sequence of events takes place during the installation process.

1. An installation menu is displayed on the screen of the target computer. The content and behavior of this menu is controlled by the base map. The person performing the installation makes selections from the menu. The installation program collects modules associated with the selected menu items until all required modules are accumulated.

2. The accumulated operating-system modules and application modules are decompressed and copied to the C partition.

3. The device-driver modules defined in the driver map are decompressed and copied to the appropriate folder in the C partition.

4. Depending on the type of image you are installing, either a complete Windows setup or Sysprep mini-setup operation begins. If you are installing an Ultra-Portable Image, a full Windows setup operation begins. If you are installing a Portable-Sysprep Image or a Hardware-Specific Image, a Sysprep-controlled mini-setup operation begins.

   Note: If you add user information to your base map, any prompts that you defined take place near the beginning of setup. If you do not add user information to your base map, prompts for specific user information occur in the normal progression of the setup process. Not taking advantage of the user-information feature might necessitate that an installer be physically present at the target computer longer.

   If you are installing an Ultra-Portable Image that uses an I386 base operating-system module, the partitioning and formatting of the drive is controlled by the primary partitioning module and the Sysprep module. The operating-system installation and basic user settings are controlled by the UNATTEND.TXT module and the ImageUltra Customizations module prepares the image for installing application modules.

5. Upon the successful completion of the operating-system setup, the setup process begins for each application program that was deployed as a module. The installation order is determined by a combination of the following:

   - The *hooks* and *slots* defined in the Options tab of attributes associated with the module
   - The Install Order tab of the base-map Settings window

Note: For more information about the Install Hook and Install Slot module attributes and the base-map Settings window, see Chapter 9, "Specifying the module-installation order," on page 167 or the ImageUltra Builder help system.

6. When all applications have been installed, the installation process is complete. If you specified in the base map that the service partition is to be deleted or resized, the operation will take place at this time.

Performing a direct-network installation

A direct-network installation combines deployment and installation into one seamless operation. Like a standard-network deployment, a direct-network installation uses either a *Network Deployment CD* or a *Network Deployment Disc* (CD or DVD) to begin the operation and select the desired base map and driver map. However, the similarities stop there:

1. As soon as the base map and driver map are selected, the person controlling the installation is immediately presented the installation menus and defines the image to be installed. This allows all operator intervention to be performed up front before any modules are copied.

2. After the menu selections have been made to define the image, a temporary network-staging folder is created on the network. Only those modules required to control the installation are unpacked to this folder.

3. All remaining modules that make up the defined image content are unpacked directly to the target computer C partition. Optionally, if console-side recovery is desired, and the base-map settings are set correctly, the complete Smart Image (or only those files required to recreate the image being installed) can be copied to the service partition.

4. Depending on the type of image being installed, either a complete Windows setup or a Sysprep mini-setup operation begins.

   Note: If you add user information to your base map, any prompts that you defined take place near the beginning of setup. If you do not add user information to your base map, prompts for specific user information occur in the normal progression of the setup process. Not taking advantage of the user-information feature might necessitate that an installer be physically present at the target computer longer.

   If you are installing an Ultra-Portable Image that uses an I386 base operating-system module, the partitioning and formatting of the drive is controlled by the primary partitioning module and the Sysprep module. The operating-system installation and basic user settings are controlled by the UNATTEND.TXT module and the ImageUltra Customizations module prepares the image for installing application modules.

5. Upon the successful completion of the operating-system setup, the setup process begins for each application program that was deployed as a module. The installation order is determined by a combination of the following:

   - The *hooks* and *slots* defined in the Options tab of attributes associated with the module
   - The Install Order tab of the base-map Settings window

   Note: For more information about the Install Hook and Install Slot module attributes and the base-map Settings window, see Chapter 9, "Specifying the module-installation order," on page 167 or the ImageUltra Builder help system.

6. When all applications have been installed, the installation process is complete. If you specified in the base map that the service partition is to be resized or deleted, the operation will take place at this time.

Advanced functions

There are two optional, advanced functions that you can use to enhance the image process: filters and utilities.

Filters

Filters automate a decision during the installation process. You typically use filters to obtain hardware-related information from a target computer to determine if a specific module or set of modules should be installed. The ImageUltra Builder program provides filters that check the machine type, machine type and model number, hardware platform (mobile or desktop), and whether or not a computer is an IBM branded computer. You can also develop your own filters if needed.

Filters are Win32-based programs that run during the installation process. In most cases filters query the BIOS of the target computer, but they can query other devices that store identification information in a ROM or EEPROM module. For example, you might want to check for the presence of a PCI modem before installing a modem-related application program or you might want to check for a specific machine type to determine which device drivers should be installed.

You can implement filters two different ways:
- Assign a filter to a menu item in either a base map or driver map by using the Filter tab attributes for the menu item.
- Assign a filter directly to an application or device-driver module by using the Filter-tab attributes for that specific module.

Using filters enables you to develop one map that works across many computer types. For information about creating your own filters or information about using the filters provided with the ImageUltra Builder program, refer to the ImageUltra Builder help system.

Utilities

Utilities are Win32-based EXE, BAT, or COM files that run during the installation process. For example, an installer might want to run CHKDSK to determine the size and condition of a hard disk before any files get installed. Utilities are copied to the service partition of the target computer during deployment, run during the installation process, but never get copied to the active partition of the target computer.

You can incorporate utilities in a map to run immediately before the installation menus, immediately after the installation menus, or as a result of a selection in the installation menus. You determine how the utilities run and the order in which they run.

The ImageUltra Builder program does not provide any utilities. If you want to include utilities, you will have to build your own utility modules.

To use a utility, you either assign a utility module to a base-map menu item or use the Before/After Menu tab in the Map Settings window for base maps.

For information about creating your own utilities, see Chapter 8, "Preparing source files for modules," on page 135. For more information about maps, see Chapter 6, "Map construction overview," on page 79 and Chapter 7, "Creating maps," on page 117.

Chapter 3. Staging-area behavior and logic

As discussed in chapter 1, the ImageUltra Builder program can use two different staging areas for deployment:

- The service partition
- The network-staging folder

In addition, some ThinkPad and ThinkCentre computers use a Hidden Protected Area (HPA) to store factory images, utilities, and diagnostic programs for recovery purposes. Although the ImageUltra Builder program does not deploy images to an HPA, there are some relationships during the deployment phase and after installation of which you might need to be aware. Also, the ImageUltra Builder program does have the ability to import modules from an HPA.

This chapter provides additional details about these staging areas and how the ImageUltra Builder program interacts with them.

Staging areas details

The information in this section provides details about the two staging areas used by the ImageUltra Builder program

The service partition

The service partition (volume label IBM_SERVICE) is a hidden partition on the hard disk that typically contains operating-system modules, application modules, device-driver modules and a set of recovery tools. It also provides its own operating environment, which has no dependency on the Windows operating system being functional or present on any active partition. Because the service partition is protected from common read and write operations, as well as most viruses, it is a practical place to store recovery, backup, and installation files and tools.

The service partition has three major purposes:

- As shipped from the factory on many older ThinkPad, NetVista, and ThinkCentre models, the service partition provides a DOS-based recovery mechanism to restore the contents of the hard disk to its factory-installed state without requiring a Windows CD or a set of *ThinkCentre* or *ThinkPad Product Recovery CDs*. The factory-installed service partition also provides a vehicle for distributing modules associated with the preinstalled software that the ImageUltra Builder program can import.
- As used by the factory-installed version of the *IBM Rescue and Recovery* product, the service partition contains its own operating environment (Windows PE) that runs independently of the Windows operating system on the C partition, a set of tools to help diagnose problems, and a set of tools to help you restore the contents of the hard disk to its factory-installed state or to one of several backed-up states. The backup and restore functions are integrated into the Rescue and Recovery product. Backup operations are from a Rescue and Recovery utility that runs in the Windows environment only. Backup files can be stored in a *virtual partition* located on the C drive, on an additional local drive, on a CD or DVD, or on a network drive, but not in the physical service partition. Restore or recovery operations can be run from either the Rescue and

Recovery utility that runs in the Windows environment, or the Rescue and Recovery Workspace that runs from the physical partition. The factory-installed recovery image is always stored in the physical service partition. The physical service partition also contains modules associated with the preinstalled software that the ImageUltra program can import.

Note: The *Rescue and Recovery* product is a standard feature on many new ThinkPad and ThinkCentre models. It is also available as a download for some other ThinkPad and ThinkCentre models at `http://www.lenovo.com/think/support/ site.wss/MIGR-4Q2QAK.html`, and is available as a retail product for non-IBM-branded computers. The factory-installed version uses a *physical service partition* for the operating environment and tools as described in this section. The user-installable versions typically use a *virtual partition* located on the C drive for their operating environment and tools.

- As used by the *ImageUltra Builder* program, the service partition is a storage area for deployed Smart Images and for the tools required to perform the image-installation process.

  When a Smart Image is deployed to a target computer that has a factory-installed Windows PE-based service partition, the Smart Image replaces the factory-installed image files (modules) with its own maps and modules. All other components are left intact.

  When a Smart Image is deployed to a target computer that has a DOS-based service partition, the ImageUltra Builder program replaces the entire DOS-based service partition with its own Windows PE-based service partition and tools.

  If a service partition does not exist on a target computer before a Smart Image is deployed, the ImageUltra Builder program creates a Windows PE-based service partition for Smart Images that are deployed through the standard network-deployment process or from *ImageUltra Distribution Discs*. For Smart Images that are deployed through a direct-network installation process, the creation of a service partition is optional.

  After a Smart Image has been deployed to the service partition and the image installed to the primary partition, the ImageUltra Builder program can deal with the Smart Image files in the service partition in one of three ways:
  – Leave the entire Smart Image intact
  – Delete all modules not used for the installed image
  – Delete the entire Smart Image

  With the latter two options, the service partition is resized or eliminated and the primary partition is resized to recover any unused space. For additional information about these settings, see "Base-map settings" on page 68 and look for the information about the Image Cleanup tab.

Access to the service partition is controlled by a boot-manager program.

The network-staging folder

The network-staging folder has one purpose; during a direct-network installation, it provides a temporary storage area on the network to unpack the maps and modules that control the installation of the image. By default, a separate network-staging folder is created for each target computer in the NETDEPL folder located under the repository being used. The name assigned to the network staging folder is based on the last 11 characters of the target computer MAC address.

Note: If necessary, you can define a location outside of the repository where the network-staging folders will be created. See "Controlling write-access to your repository during a direct-network install" on page 181 for more information.

Unlike deployment to a service partition, the person at the target computer who is controlling the direct-network deployment makes choices from the installation menu system *before* any modules are sent to the target computer. Then, only those modules required for the selected image are obtained from the repository, unpacked dynamically, and installed on the target computer. The target computer must remain connected to the network throughout the entire deployment and installation process. When the installation process is complete, the network-staging folder is automatically erased.

A direct-network deployment does not require that a service partition be present on the target computer. The use of a service partition is optional, and if used, its only purpose is to store maps and modules for client-side recovery. A service partition is not used as a staging area during a direct-network installation process.

- If you do not want maps and modules copied to a service partition, you minimize the deployment time, but you will not have client-side recovery for the image you deployed.
- If you want only those maps and modules that are required to recreate the installed image copied to a service partition, deployment time will be longer, but you will have client-side recovery for the specific image that is installed on the target computer.
- If you want the complete Smart Image copied to a service partition, deployment time will be longer yet, but you will have client-side recovery for any of the images that can be installed from the Smart Image.

For additional information about these settings, see "Base-map settings" on page 68, and look for the information about the Image Cleanup tab.

Deployment and installation overview

This section provides an overview of the various deployment and installation processes.

Standard network-deployment process with local installation

The following illustration shows the standard network-deployment process with local installation. Notice that the service partition is used as the staging area for all maps and modules required for the Smart Image.
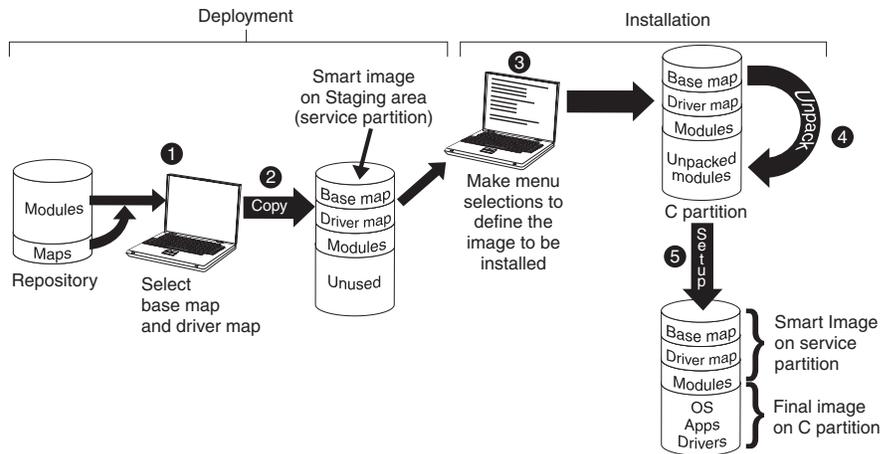
1. The person controlling the deployment starts the process by booting the target computer with an *ImageUltra Builder Deployment Disc* (CD or DVD) or an *ImageUltra Builder Deployment CD*, and then selecting the required maps from the repository.

2. The selected maps and all modules defined within the maps are obtained from the repository and copied over the network to the service partition, which acts as the staging area for the complete Smart Image.

   Note: At this point, a message is displayed stating that the deployment phase is complete. The person controlling the deployment can either shut down the computer at this point and ship it to another location, or continue with the installation phase.

3. The person controlling the installation process, starts the installation phase by making selections from the installation menus to define the image to be installed.

4. The modules required for the defined image are unpacked from the service partition onto the C partition.

5. The setup process begins. When the installation phase is over, the C partition contains the new image and the service partition contains the Smart Image, which can be used for client-side recovery.

   Note: An optional cleanup step can be defined in the base map Map Settings window to minimize the size of the service partition and keep only those files required for the installed image, or eliminate the service partition.

CD or DVD distribution

The following illustration shows the deployment process using an *Image Ultra Builder Distribution Disc* (CD or DVD) and a local-installation process, which uses a service partition as a staging area.

Deployment Installation

Smart image on Staging area (service partition)

❶ Create distribution CD

❷ Start target computer with distribution CD

❸ Copy

Base map / Driver map / Modules / Unused

❹ Make menu selections to define the image to be installed

Base map / Driver map / Modules / Unpacked modules — C partition — Unpack ❺

❻ Setup

Base map / Driver map / Modules / OS Apps Drivers — Smart Image on service partition / Final image on C partition

Modules / Maps — Repository

1. The first step in CD or DVD deployment is the creation of a set of *ImageUltra Builder Distribution Discs*. (See the ImageUltra Builder help system for details).
2. The person controlling the deployment starts the process by booting the target computer with an *ImageUltra Builder Distribution Disc*.
3. The maps and modules contained on the set of discs are copied to the service partition, which acts as the staging area for the complete Smart Image.

   Note: At this point, a message is displayed stating that the deployment phase is complete. The person controlling the deployment can either shut down the computer at this point and ship it to another location, or continue with the installation phase.

4. The person controlling the installation process, starts the installation phase by making selections from the installation menus to define the image to be installed.
5. The modules required for the defined image are unpacked from the service partition onto the C partition.
6. The setup process begins. When the installation phase is over, the C partition contains the new image and the service partition contains the Smart Image, which can be used for client-side recovery.

   Note: An optional cleanup step can be defined in the base map Map Settings window to minimize the size of the service partition and keep only those files required for the installed image, or eliminate the service partition.

Direct-network installation

The following illustration shows a direct-network installation, which uses a network-staging folder as a staging area.

Copy (optional)
Unpack

Base map / Driver map / Modules / OS Apps Drivers — (Optional) Smart Image or copy of installed image / Final image on C partition

Modules / Maps — Repository

❶ Select base map and driver map

Unpack — Unpack required modules ❷

Base map / Driver map / Unpacked installation modules — Staging area (Network-staging folder)

❸ Make menu selections to define the image to be installed

Setup ❹
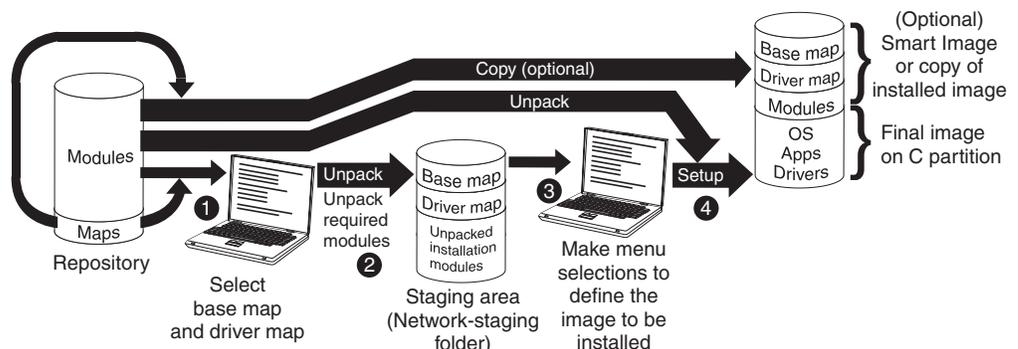
1. The person controlling the direct-network installation starts the process by booting the target computer with an *ImageUltra Builder Deployment Disc* (CD or DVD) or an *ImageUltra Builder Deployment CD*, and then selects the required maps from the repository.

2. A network-staging folder is created specifically for the target computer. The selected maps and a limited number of modules (only those required to control the installation process) are copied to the network-staging folder, which acts as a staging area for the installation process.

3. The person controlling the direct-network installation makes selections from the installation menus to define the image to be installed.

4. The modules required for the defined image are obtained directly from the repository and copied over the network to target computer C partition. Then, the setup process begins.

   Note: Optionally, if a developer of the image specified that a service partition is to be created for client-side recovery purposes, the maps and modules also are copied to a service partition.

   When setup is complete, the C partition contains the new image.

ImageUltra Builder relationships with the Hidden Protected Area (HPA)

This section provides information about the purpose of the Hidden Protected Area (HPA) and the relationships that exist between the HPA and the ImageUltra Builder program. Additional information about the HPA is available from the Lenovo Web site at:
http://www.lenovo.com/think/support/site.wss/ document.do?lndocid=MIGR-46023

What is a Hidden Protected Area?

Implemented in 2003, the Hidden Protected Area (HPA) is a variant in the IBM disk-to-disk recovery scheme. Like the service partition, the HPA provides its own operating environment and contains operating-system modules, application modules, device-driver modules, and a set of recovery tools. But, the similarities stop there. The architecture of the HPA provides a secure environment. Protection settings (security levels) and access to the HPA are controlled by the computer basic input/output system (BIOS). The HPA does not rely on a boot manager, and is therefore immune from boot-sector viruses, which could cripple other disk-to-disk recovery methods.

Each application within the HPA is contained on its own protected service area (PSA). There are two types of PSAs: Boot PSAs and Data PSAs. Boot PSAs contain all of the files necessary to boot and start the application from the associated Data PSA and to display an icon on a secure launching area called the Access IBM Predesktop Area. Each Boot PSA is digitally signed to deter tampering and to deter viruses. Each time a Boot PSA is used, the signature is checked. Only those PSAs with valid signatures are permitted to boot. Data PSAs contain all of the files associated with the application itself.

Some ThinkCentre and ThinkPad models, the HPA provides the tools to diagnose a problem and recover from a catastrophic failure by restoring the contents of the hard disk to a factory-installed state. Access to these tools is provided through the IBM Predesktop Area.

Recovery impacts

Because the content of the HPA is not affected by the ImageUltra Builder deployment or installation processes, it is very likely that you can have multiple forms of recovery after a Smart Image has been installed on a target computer that has an HPA. The presence or absence of the Rescue and Recovery product also has an impact on recovery options.

Because the tools within the HPA are DOS based and the HPA is protected against alterations, when the ImageUltra Builder program is installed on a computer that has an HPA, the ImageUltra Builder program creates its own service partition and installs a boot manager to bypass the IBM Predesktop Area. When F11 is pressed during startup, instead of booting to the IBM Predesktop Area, the computer boots to service partition and opens the ThinkVantage ImageUltra Builder workspace. All recovery operations are performed from the ThinkVantage ImageUltra Builder workspace.

If a user selects **Restore your system --> Restore my hard drive to the original factory state** from the ThinkVantage ImageUltra Builder workspace menu, the ImageUltra Builder installation menu system opens, which enables the user to reinstall the Smart Image instead of the original factory-installed image.

Importing modules and maps from an HPA

The ImageUltra Builder program can import modules and maps from an HPA just like it does from a service partition. For more information about importing modules, see "Importing maps and modules from a service partition or HPA" on page 14.

Using drivers in an HPA

Because the contents of an HPA are preserved while a Smart Image is being installed, the ImageUltra Builder program can use the device drivers stored in the HPA if a driver map is not defined for the Smart Image. For more information, see "Importing maps and modules from a service partition or HPA" on page 14.

ImageUltra Builder relationships with the Rescue and Recovery product

This section provides information about the Rescue and Recovery product and the relationships that exist between the Rescue and Recovery product and the ImageUltra Builder program.

What is the Rescue and Recovery product?

The Rescue and Recovery product provides a set of tools to help users diagnose, get help, and recover from a system failure even if the Windows operating system on the C partition cannot be started. The Rescue and Recovery product is available as follows:

- As a standard, preinstalled feature on many ThinkPad and ThinkCentre computers manufactured since 2003.
- As a retail product for non-IBM-branded computers
- As a downloadable product for selected ThinkPad and ThinkCentre computers from

  http://www.ibm.com/pc/support/site.wss/MIGR-4Q2QAK.html

The Rescue and Recovery tools reside in two locations:

- On the Windows side (C partition) are a set of tools that enable a user to create backups, rescue media, and recovery media.
- In a Windows PE-based partition is the Rescue and Recovery workspace, an environment that runs independently of the Windows operating system and is hidden from the Windows operating system.

  Because the Rescue and Recovery workspace is hidden, it is immune from most types of virus attacks and provides a safe place from which to perform rescue and recovery operations that might not be possible in the Windows environment.

The factory-installed version of the Rescue and Recovery product uses a service partition. The user-installable versions of the Rescue and Recovery product typically installs in and uses a virtual partition. However, if the user-installable versions of the Rescue and Recovery products detect a preexisting Windows PE-based service partition with at least 500MB of free space during the installation, it will install in and use the service partition.

Impacts

As of the publishing of this book, there are three versions of the Rescue and Recovery program:
- Rescue and Recovery with Rapid Restore (referred to in this book as Rescue and Recovery version 1)
- Rescue and Recovery - Rapid Restore (referred to in this book as Rescue and Recovery version 2)
- Rescue and Recovery 3 (referred to in this book as Rescue and Recovery version 3)

Because of the variations among these versions, the relationship between each version and the ImageUltra Builder program vary.

The following scenarios are provided to help you understand the relationship between the ImageUltra Builder program and the various versions of the Rescue and Recovery product.
- **Scenario 1:** Installing Rescue and Recovery version 3 on a computer that has a Smart Image installed. In this scenario, if a service partition already exists, the content of the service partition will remain mostly unchanged. The name of the ThinkVantage ImageUltra Builder workspace will change to the Rescue and Recovery workspace and the list of menu items might be extended to provide more recovery options.

  If a service partition does not already exist, the Rescue and Recovery version 3 program will be installed in a virtual partition.

  Additionally, whether or not a service partition already exists, several additional Rescue and Recovery tools will be installed on the C partition and added to the Windows Start menu.
- **Scenario 2:** Installing Rescue and Recovery versions 1 or 2 on a computer that has a Smart Image installed. This scenario is not supported if a service partition already exists. If a service partition does not already exist, the Rescue and Recovery version 1 or 2 program will be installed in a virtual partition and several additional Rescue and Recovery tools will be installed on the C partition and added to the Windows Start menu.
- **Scenario 3:** Installing a Smart Image on a target computer that has a service partition containing the Rescue and Recovery version 1 or 2 program. In this

scenario, only the maps and modules in the service partition get replaced. No
other modifications are made to the service partition contents.

Security dependencies

The ImageUltra Builder program is able to use the device-driver modules from the
HPA during deployment if a driver map is not used. However, in order to do so,
the HPA security settings in the target computer BIOS settings must be set to either
High security or **Medium security**. If the security setting is set to **Security
disabled**, the ImageUltra Builder program does not attempt to use the
device-driver modules from the HPA; instead, it will check to see if a service
partition exists and attempt to use the device-driver modules from the service
partition.

On computers that are not HPA-enabled, there are no corresponding BIOS security
settings. If a Smart Image is deployed without a driver map, the ImageUltra
Builder program always attempts to use the device drivers from the service
partition.

Staging-area logic

Because various conditions might exist on different target computers, the
ImageUltra Builder program uses built-in logic to ensure that a common Smart
Image can be used across a wide range of computers. Varying conditions include:

- Support or the lack of support for an HPA in the BIOS
- The presence or absence of an HPA
- The presence or absence of a service partition

The following table shows how the deployment behavior changes on an IBM
branded computer based on conditions that exist before the image is deployed.

Notes:

1. The use of driver maps and the use of device driver modules from either the
   service partition or HPA is limited to Portable-Sysprep Images and
   Ultra-Portable Images.
2. You also can use the following table to determine the behavior for a
   direct-network install that specifies the use of a service partition for
   console-side recovery purposes.

Table 4. Staging-area behavior - IBM branded computers

	Conditions on an IBM branded target computer prior to deployment			
Maps deployed	HPA support in BIOS	HPA present with security settings set to Medium or High	Service partition present	Result from deployment
Base map and driver map	Yes	No	Yes	• Create the service partition. • Deploy the Smart Image as defined in base and driver maps to the service partition.

Table 4. Staging-area behavior - IBM branded computers (continued)

| Maps deployed | Conditions on an IBM branded target computer prior to deployment | | | Result from deployment |
	HPA support in BIOS	HPA present with security settings set to Medium or High	Service partition present	
Base map and driver map	No	No	No	• Create the service partition. • Deploy the Smart Image as defined in the base and driver maps to the service partition.
Base map and driver map	Yes	Yes	No	• Create the service partition. • Deploy the Smart Image as defined in the base and driver maps to the service partition.
Base map and driver map	Yes	No	No	• Delete all unnecessary modules and files from the service partition. Preserve any Rapid Restore files or Rescue and Recovery files that might exist. • Resize the service partition, if required. • Deploy the Smart Image as defined by the base and driver maps to the service partition.
Base map only	Yes	No	Yes	• Create the service partition. • Deploy all modules defined in the base map to the service partition. **Note:** If a driver map is not used under these conditions, setup will use the native Windows device drivers and some features might not be enabled or optimized.

Table 4. Staging-area behavior - IBM branded computers (continued)

	Conditions on an IBM branded target computer prior to deployment			
Maps deployed	HPA support in BIOS	HPA present with security settings set to Medium or High	Service partition present	Result from deployment
Base map only	No	No	No	• Create the service partition. • Deploy all modules defined in the base map to the service partition. **Note:** If a driver map is not used under these conditions, setup will use the native Windows device drivers and some features might not be enabled or optimized.
Base map only	Yes	Yes	No	• Create the service partition. • Deploy all modules defined in the base map to the service partition. • Use device-driver modules from the HPA and use them during installation.
Base map only	Yes	No	Yes	• Delete all unnecessary modules and files from the service partition. Preserve any Rescue and Recovery files that might exist. • Resize the service partition, if required. • Deploy the Smart Image as defined by the base map to the service partition. • Use device-driver modules that are currently in the service partition. **Note:** If a driver map is not used under these conditions, and no device drivers are in the existing service partition, setup will use the native Windows device drivers and some features might not be enabled or optimized.

Table 4. Staging-area behavior - IBM branded computers (continued)

Maps deployed	Conditions on an IBM branded target computer prior to deployment			Result from deployment
	HPA support in BIOS	HPA present with security settings set to Medium or High	Service partition present	
Base map only	Yes	Yes	Yes	• Delete all unnecessary modules and files from the service partition. Preserve any Rescue and Recovery files that might exist. • Resize the service partition, if required. • Deploy the Smart Image as defined by the base map to the service partition. • Source the device driver modules from the HPA and use them during installation.

When using the ImageUltra Builder program on a non-IBM branded computer, the presence or absence of a pre-existing service partition effects what takes place during deployment. A service partition on a non-IBM branded computer is typically the result of a previous deployment of a Smart Image using the *ImageUltra Builder* program or the installation of the *Rescue and Recovery* retail product. If a Smart Image is deployed to a non-IBM branded computer, the Rescue and Recovery product and any backup files stored locally on the primary hard disk will be lost.

The following table shows how the deployment behavior changes on a non-IBM branded computer based on the conditions that exist before an image is deployed.

Notes:

1. On non-IBM branded computers, the use of driver maps and the sourcing of device-driver modules from an existing service partition is limited to Portable-Sysprep Images and Ultra-Portable Images that use an I386 base operating-system module.

2. You also can use the following table to determine the behavior for a direct-network install that specifies the use of a service partition for console-side recovery purposes.

Table 5. Staging area behavior - non-IBM branded computers

Maps deployed	Service partition present	Result from deployment
Base map and driver map	No	• Create a service partition. • Deploy the Smart Image as defined in the base and driver maps to the service partition.

Table 5. Staging area behavior - non-IBM branded computers (continued)

Maps deployed	Service partition present	Result from deployment
Base map only	Yes	• Delete all unnecessary modules and files from the service partition. • Resize the service partition, if required. • Convert the services partitions to WindowsPE, if required. • Deploy the modules defined in the base map. • Use the device driver modules in the existing service partition. **Note:** If a driver map is not used under these conditions, and no device drivers are in the existing service partition, setup will use the native Windows device drivers and some features might not be enabled or optimized.

Chapter 4. Installing the ImageUltra Builder program

Before using the information in this chapter, you must have a good understanding of the image-creation process and be familiar with some of the terminology associated with the ImageUltra Builder program. Take the time to read Chapter 1, "ImageUltra Builder capabilities," on page 1 and Chapter 2, "An overview of the image process," on page 11.

Then, review the minimum requirements, prerequisites, considerations, and limitations in this chapter before you install the ImageUltra Builder program.

Minimum requirements

The following information provides the minimum requirements for the ImageUltra Builder console, server, and the target computers.

ImageUltra Builder server computers

Computers intended to be used as ImageUltra Builder servers, must meet the following minimum requirements:

Operating system: Microsoft® Windows® XP Professional Edition or Windows 2000

Note: MDAC 2.7 or higher is required for ImageUltra Builder. It is included in Microsoft® Windows® XP Professional, but not in Windows 2000. If you are using Microsoft® Windows® 2000, you need to download Microsoft Data Access Components (MDAC) 2.7 or higher from Microsoft® at:

`http://www.msdn.microsoft.com/data/mdac/downloads/default.aspx`

Disk Space: 200 MB of available disk space to install the ImageUltra Builder program

Repository: The amount of disk space required for the repository depends on the size and number of modules that will be stored in it.

Networking requirements: The server computer must be network enabled for a direct-network installation or a standard network deployment with local installation. There are no networking requirements if images are installed from *Distribution Discs*.

ImageUltra Builder console computers

Computers intended to be used as ImageUltra Builder consoles, must meet the following minimum requirements:

Operating system: Microsoft® Windows® XP Professional Edition or Windows 2000

Note: MDAC 2.7 or higher is required for ImageUltra Builder. It is included in Microsoft® Windows® XP Professional, but not in Windows 2000. If you are using Microsoft® Windows® 2000, you need to download Microsoft Data Access Components (MDAC) 2.7 or higher from Microsoft® at:

`http://www.msdn.microsoft.com/data/mdac/downloads/default.aspx`

Disk Space: 400 MB of available disk space to install the ImageUltra Builder program

Repository: The amount of disk space required for the repository depends on the size and number of modules that will be stored in it.

Removable-media drives:
- CD or DVD drive for installation
- Recordable CD or DVD drive for the creation of *Network Deployment Discs* or *Distribution Discs*

Minimum networking speed: 10 Mbps; Ethernet or token ring

ImageUltra Builder target computers

Target computers must meet the following minimum requirements before you deploy Smart Images to them:

Operating system: The target computer must meet the minimum requirements to install and run the operating system being deployed.

Networking requirements: The target computer must be network enabled for a direct-network installation or a standard network deployment with local installation. There are no networking requirements if images are installed from *Distribution Discs*.

Removable-media drives: The target computer must have a bootable, removable-media drive (CD or DVD drive) to accommodate a *Network Deployment Disc*, or *Distribution Disc*.

Note: USB, CD and DVD drives are not supported for use with *Network Deployment Discs*, or *Distribution Discs*.

Prerequisites

The following are installation prerequisites for the ImageUltra Builder Version 3.0 program to perform properly, and for you to create and distribute Smart Images legally with the proper licensing:

- The repository must be accessible to the ImageUltra console, the source computer for import operations, and any target computer that will have a Smart Image deployed over the network.
- Ensure proper licensing:
  - The ImageUltra Builder license agreement applies to the ImageUltra Builder program only. Read the license carefully during the installation process. All other licenses, warranty service, and support for non-Lenovo products, such as Microsoft Sysprep, Symantec Norton Ghost, and PowerQuest DeployCenter, must be acquired from the third party, not Lenovo. Lenovo makes no representations or warranties regarding non-Lenovo products. If you need to review the license after installation, click **Help** in the ImageUltra Builder menu bar, and then click **View the license**.
  - Lenovo requires that you obtain a per-seat license for each computer on which a Smart Image is deployed. For information about ordering licenses, contact your Lenovo marketing representative.
  - You are solely responsible for obtaining any licenses required to distribute operating systems, application programs, and any other licensed software.

Installation considerations

Consider the following information when installing the ImageUltra Builder program.

The ImageUltra server installs IBM® Cloudscape™ V10.0, a relational database program that is written entirely in Java and is supported on any certified Java Virtual Machine (JVM). Running Cloudscape requires a certified JVM and at least 4MB of available memory.

The repository can be installed on the following computers:

- ImageUltra Builder server
- ImageUltra Builder console
- Shared disk of another computer (where the server or console program is installed)
- The computer intended to be used as both the console and server computer.

Notes:

1. Additional storage is required for repositories. So make sure you have enough storage for future growth.
2. The repository must be accessible to the ImageUltra console, the source computer for import operations, and any target computer that will have a Smart Image deployed over the network.

- When building driver modules, use only certified Microsoft WHQL device drivers. Device drivers must have an.INF file extension to install correctly. Building modules with unsigned device drivers might cause Windows to display information messages during installation, thereby preventing an image from installing in an unattended manner.
- Research compatibility between operating systems and versions of Symantec Norton Ghost and PowerQuest DeployCenter before you begin creating Portable-Sysprep Images or Hardware-Specific Images.
- If you intend to use stand-alone *Distribution Discs* or *Network Deployment Discs*, the ImageUltra Builder program can create CDs.
- One repository is sufficient for most environments, you can create additional repositories using the New Repository Wizard. Before creating multiple repositories, it is a good idea to develop a repository strategy. Consider the following:
  - When you create a new repository, make sure you assign it a unique ID that has not been used for any other repository in your company. If you have duplicate IDs, and import modules from a source computer that has a Smart Image that was deployed from a different repository, there is a chance that modules can get overwritten.
  - All modules defined in a map must be in the same repository as the map.
  - The driver map and base map used for deployment must be in the same repository.
- Consider using a separate server to back up your module repository.
- It is possible to create an image that contains modules that collide, or fail to interact correctly.

  For example, an image that contains two antivirus program modules might not install correctly. As a result, the image might not work. One way to prevent the failure is to read the information provided with devices, device drivers, and applications before beginning to create modules.

Limitations

The following limitations apply to the functionality of the ImageUltra Builder program:

- To deploy an Ultra-Portable Image without a driver map, the target computer must have a service partition or Hidden Protected Area (HPA).
- Ultra-Portable Images and Portable-Sysprep Images can be based on Windows 2000 and Windows XP only. Hardware-Specific Images can be based on any Windows operating system.

Installing the ImageUltra Builder 3.0 program

If you are upgrading from a previous version of the ImageUltra Builder program, use the procedure in "Upgrading from a previous version of ImageUltra Builder" on page 53.If you are installing ImageUltra Builder for the first time, follow one of the three procedures below.

Installing an ImageUltra Builder console

This is the default installation setting. To install ImageUltra Builder on a console computer do the following:

1. Insert the ImageUltra Builder CD into the CD drive of the computer. In most cases, the install program will start automatically. If the install program does not start automatically, complete the following procedure:
   a. Click **Start**.
   b. Click **Run**.
   c. Type *d*:\SETUP.EXE where *d* is the letter of your CD drive, then click **OK**. The ImageUltra Builder InstallShield Wizard opens.
2. After the welcome screen has opened, click **Next**.
3. When the License Agreement appears, read the terms, then select the radio button next to 'I accept the terms in the license agreement', or 'I do not accept the terms in the license agreement.' Selecting accept the terms in the license agreement.' will continue on to the next screen.
4. In the next screen, type your User Name and Organization.
5. Select the appropriate radio button under 'Install this application for: ', then click **Next**.
6. Select ImageUltra Builder Console, and make sure that ImageUltra Builder server is deselected.

   To select ImageUltra Builder console and deselect ImageUltra Builder Server do the following:
   a. Select the drop-down menu next to ImageUltra Builder Console, then select 'This feature, and all sub features, will be installed on local hard drive'.
   b. Select the drop-down menu next to ImageUltra Builder Server, then select 'This feature will not be available'.
7. Click **Next**.
8. The 'Ready to Install the Program' screen will appear.
9. Click **Install**.
10. When the Installation is complete, click **Finish**. The ImageUltra Builder shortcut has been installed on your desktop, and you can now begin using ImageUltra Builder.

Note: If you install ImageUltra Builder on a console, you must also follow the steps to install ImageUltra Builder on a server.

Installing an ImageUltra Builder server

If you have installed ImageUltra Builder on a console, you must now also install it on a server. To install ImageUltra Builder on a Server, do the following:

1. Insert the ImageUltra Builder CD into the CD drive of the computer. In most cases, the install program will start automatically. If the install program does not start automatically, complete the following procedure:

   a. Click **Start**.

   b. Click **Run**.

   c. Type d:\SETUP.EXE where d is the letter of your CD drive, then click **OK**. The ImageUltra Builder InstallShield Wizard opens.

2. After the welcome screen has opened, click **Next**.

3. If you have not already installed ImageUltra Builder Console, continue with step 4. If you have installed ImageUltra Builder console, complete the following procedure:

   a. The program Maintenance Screen will appear. Select the radio button next to Modify, then click **Next**.

   b. Continue with step 6.

4. When the License Agreement appears, read the terms, then select the radio button next to 'I accept the terms in the license agreement.', or 'I do not accept the terms in the license agreement.' Selecting 'I accept the terms in the license agreement.' will continue on to the next screen.

5. In the next screen, type your User Name and Organization.

6. Select the appropriate radio button under 'Install this application for: ', then click **Next**.

7. Select ImageUltra Builder Server, and make sure that ImageUltra Builder Console is deselected.

   To select ImageUltra Builder Server and deselect ImageUltra Builder Console do the following:

   a. Select the drop-down menu next to ImageUltra Builder Server, then select 'This feature, and all sub features, will be installed on local hard drive'.

   b. Select the drop-down menu next to ImageUltra Builder Console, then select 'This feature will not be available'.

8. Click **Next**.

9. The 'Secure Server Repositories' screen will appear. Select the appropriate radio button next to Yes or No, then click **Next**.

   Note: If you answer Yes, the Installation wizard will automatically add a default Administrator ID and Password which will need to be changed. For information on how to change the Administrator password, see "Changing the default administration password" on page 52.

10. The 'Ready to Install the Program' screen will appear. Click **Install**.

11. When the installation is complete, click **Finish**.

Accessing the ImageUltra Builder server

To access the ImageUltra Builder server do the following:

1. Open Windows Explorer.

2. Open the Program files folder on your main drive.

3. Select the **IBM folder**, then select the **Cloudscape_10.0 folder**.

4. Double-click **StartCloudscape.cmd**.

Installing an ImageUltra Builder console and server

If you are not familiar with the ImageUltra Builder program, then you might want to install it with both the console and server on the same computer. To install ImageUltra Builder with both the console and server on the same computer, do the following:

1. Insert the ImageUltra Builder CD into the CD drive of the computer. In most cases, the install program will start automatically. If the install program does not start automatically, complete the following procedure:

   a. Click **Start**.

   b. Click **Run**.

   c. Type *d*:\SETUP.EXE where *d* is the letter of your CD drive, then click **OK**. The ImageUltra Builder InstallShield Wizard opens.

2. After the welcome screen has opened, click **Next**.

3. When the License Agreement appears, read the terms, then select the radio button next to 'I accept the terms in the license agreement.', or 'I do not accept the terms in the license agreement.' Selecting 'I accept the terms in the license agreement.' will continue on to the next screen.

4. In the next screen, type your User Name and Organization.

5. Select the appropriate radio button under 'Install this application for:', then click **Next**.

6. Select both ImageUltra Builder Console and ImageUltra Builder Server.

   To select ImageUltra Builder Console and ImageUltra Builder Server do the following:

   a. Select the drop-down menu next to ImageUltra Builder Console, then select 'This feature, and all sub features, will be installed on local hard drive'.

   b. Select the drop-down menu next to ImageUltra Builder Server, then select 'This feature, and all sub features, will be installed on local hard drive'.

7. Click **Next**.

8. The 'Secure Server Repositories' screen will appear. Select the appropriate radio button next to Yes or No, then click **Next**.

   Note: If you answer Yes, the Installation wizard will automatically add a default Administrator ID and Password which will need to be changed. For information on how to change the Administrator password, see "Changing the default administration password."

9. The 'Ready to Install the Program' screen will appear. Click **Install**.

10. When the installation is complete, click **Finish**.

Changing the default administration password

If you chose to make the server secure, an Administrator ID and password were automatically generated for you. The default ID is **Admin** and the default password is **pwd**. To change the password:

1. Open ImageUltra Builder.

2. Make sure the ImageUltra Builder Server is running.

   To access the ImageUltra Builder Server:

   a. Open Windows Explorer.

b. Open the Program files folder on your main drive.

c. Select the **IBM folder**, then select the **Cloudscape_10.0 folder**.

d. Double-click **StartCloudscape.cmd**.

3. Select Tools from the top menu bar, then click Administration from the drop down menu.

4. Select or type in the ImageUltra server name that you want to log on to.

5. Make sure that the User name is correct *Admin*, then type the default password *pwd*.

6. Click **Login**, then click **Administer users** from the 'What would you like to do?' menu.

7. Click **Change a user**, then click **Admin**, then click **Change the password**.

8. Type a new password that is less than 20 characters long in the 'Type a new password:' box, then retype the password in the 'Type the new password again to confirm:' box.

   Note: Passwords are case-sensitive.

9. Click **Change Password**.

10. The 'What do you want to change about Admin?' screen will reappear. Your password has been successfully changed.

Upgrading from a previous version of ImageUltra Builder

Use the following procedure to upgrade from a previous version of the ImageUltra Builder program:

Note: To ensure that repositories created using earlier versions of ImageUltra Builder can run on ImageUltra Builder Version 3.0, be sure to install Cloudscape somewhere on your computer.

1. Insert the ImageUltra Builder CD into the CD drive of the computer that will be used as the ImageUltra Builder console. In most cases the install program will start automatically. If the install program does not start automatically, complete the following procedure:

   a. Click **Start**.

   b. Click **Run**.

   c. Type *d***:\SETUP.EXE** (where *d* is the letter of your CD drive).

   d. Click **OK**.

2. Follow the instructions on the screen.

   Note: No files from the previous version are overwritten by the installation process. The new version of the ImageUltra Builder program is installed in its own folder and the ImageUltra Builder 3.0 icon is added to the Windows desktop.

3. When the ImageUltra Builder program installation is complete, click the **ImageUltra Builder 3.0** icon.

4. Click **File**, then click **Migrate**.

5. Click **Next**.

6. In the **Location** field, type the path for the previous repository that you want to migrate, including the name of the folder, or use the **Browse** button to find a location.

7. In the **Server** field, type the full name of the server on which to create the repository.

8. Click **Next**.

9. In the **Description** field, type a descriptive name or phrase for the repository you want to migrate.

10. Click **Next**

11. Review your new repository settings. To complete the migrate repository process, click **Finish**. To revise a setting, click **Back**.

Getting help after installation

The ImageUltra Builder program has a built-in help system to assist you with most on-screen tasks. You can access the help system using any of the following methods:

- Click the **Help** button provided on the Main window tool bar.
- Click **Help** on the Main window menu bar, and then click **View help topics**.
- Press **F1**.

If you need additional assistance, see Chapter 10, "Getting help and support," on page 173.

Chapter 5. Working with the ImageUltra Builder interface

The ImageUltra Builder interface provides the following important functions:

- It enables you to manipulate modules within a repository, create additional repositories, and navigate among all defined repositories;
- It provides direct access to tools used to import, build, and promote modules, and also to deploy Smart Images;
- It enables you to develop base maps and device-driver maps;

  And now, maps are displayed under Windows PE, so they provide a more attractive, intuitive work area in which to deploy Smart Images.

This chapter explains the major components of the Image Ultra Builder interface, where the components are located, and how to navigate within the interface.

Conventions used within the interface

The following table displays the various icons used in the ImageUltra Builder program.

Table 6. Icons used within the interface

Icon	Name	Description	Keyboard Shortcut
	Save	Saves changes in the active map or module window.	Ctrl+S
	Cut	Cuts the currently selected object.	Ctrl+X
	Copy	Copies the currently selected object to the clipboard.	Ctrl+C
	Paste	Pastes an object from the clipboard.	Ctrl+V
	Build module	Builds the currently selected module.	
	Promote	Promotes the currently selected map or module to the gold state.	
	New Operating-system Module	Launches the New Module Wizard to create new operating system modules. This icon is available in the Repository window toolbar only when the Operating Systems tab is selected in the Repository window.	
	New Base Map	Launches the New Map Wizard to create new base maps. This icon is available in the Repository window toolbar only when the Base Maps tab is selected in the Repository window.	

Table 6. Icons used within the interface (continued)

Icon	Name	Description	Keyboard Shortcut
	New Driver Map	Launches the New Map Wizard to create new driver maps. This icon is available in the Repository window toolbar only when the Driver Maps tab is selected in the Repository window.	
	New Application Module	Launches the New Module Wizard to create new application modules. This icon is available in the Repository window toolbar only when the Applications tab is selected in the Repository window.	
	New Device Driver Module	Launches the New Module Wizard to create new device driver modules. This icon is available in the Repository window toolbar only when the Device Drivers tab is selected in the Repository window.	
	New Filter Module	Launches the New Module Wizard to create new filter modules.This icon is available in the Repository window toolbar only when the Filters tab is selected in the Repository window.	
	New Utility Module	Launches the New Module Wizard to create new utility modules. This icon is available in the Repository window toolbar only when the Utilities tab is selected in the Repository window.	
	New Repository	Launches the New repository Wizard to create a new repository. This icon is available in the Main window toolbar when the Repository window is active.	

Table 6. Icons used within the interface (continued)

Icon	Name	Description	Keyboard Shortcut
	Insert Filter	The function of this icon varies based on where the icon is selected: • When selected from the Main window toolbar, this icon associates a filter with a base-map or driver map menu item. The icon appears on the Main window toolbar only when a map is open. • When selected from the Filters tab of an operating-system module, application module, or device-driver module; this icons assigns a filter to the selected module.	
	Import	Launches the Import Wizard to import modules or maps from source computers, CDs, folders, ImageUltra export packages, and other repositories.	
	Export	Launches the Export Wizard to backup maps and modules to an archive folder.	
	Deploy	Launches the Deploy Wizard to create a *Network Deployment CD*, *Network Deployment CD*, or standalone *Distribution CDs*.	
	Network Sync Settings	Opens the Network Sync Settings window so you can define replacement maps for maps that have already been deployed.	
	Open Repository	Enables you to open a repository or change to a different repository.	Ctrl+O
	Delete	Deletes the selected map, module, menu item, or link.	Del
	Views	Enables you to change the view of the repository to use large icons, small icons, a list, or details.	
	Move up	Moves a menu item up in base maps and driver maps	Alt+Up Arrow
	Move down	Moves a menu item down in base maps and driver maps	Alt+Down Arrow
	Base map	Used in the Repository window to identify an object as a base-map. When this icon is gold, the base map is in the gold state.	

Table 6. Icons used within the interface (continued)

Icon	Name	Description	Keyboard Shortcut
	Driver map	Used in the repository Window to identify an object as a driver map. When this icon is gold, the driver map is in the gold state.	
	Operating-system module	Used in the Repository window, Base Map window, and Driver Map window to identify an object as an operating-system module. When this icon is gold, the operating-system module is in the gold state.	
	Application module	Used in the Repository window, Base Map window, and Driver Map window to identify an object as an application module. When this icon is gold, the application module is in the gold state.	
	Device-driver module	Used in the Repository window and Driver Map window to identify an object as a device-driver module. When this icon is gold, the device-driver module is in the gold state.	
	Filter module	Used in the Repository window, Base Map window, and Driver Map window to identify an object as a filter module. When this icon is gold, the filter module is in the gold state. When a filter is associated with a menu item or other module, a miniature version of this icon overlays the menu-item icon or module icon.	
	Utility module	Used in the Repository window and Base Map window to identify an object as a utility module. When this icon is gold, the utility module is in the gold state. When a utility is associated with a menu item or other module, a miniature version of this icon overlays the menu-item icon or module icon.	
	Help	Opens the help system.	F1

Table 6. Icons used within the interface (continued)

Icon	Name	Description	Keyboard Shortcut
	Container	Used in the Repository window, Base Map window, and Driver Map window to identify an object as a container. This icon overlays other module icons. For example, if this icon overlays a device-driver module icon, that object is a device-driver container.	
	Menu item	Used in Base Map windows and Driver Map windows to identify an object as a menu item.	
	Link	Used in Base Map windows to identify an object as a link.	
	Insert Menu Item	Inserts a menu item in a base map or driver map. This icon appears in the Main window toolbar only when a Base Map window or Driver Map window is open.	
	Insert Operating System	Inserts an operating system module under a selected item in a base map. This icon appears in the Main window toolbar only when a Base Map window or Driver Map window is open.	
	Insert Application	Inserts an application module under a selected item in a base map. This icon appears in the Main window toolbar only when a Base Map window or Driver Map window is open.	
	Insert Device Driver	Inserts a device-driver module under a selected item in a driver map. This icon appears in the Main window toolbar only when a Base Map window or Driver Map window is open.	
	Insert Utility	Associates a utility module with the selected item in a base map. This icon appears in the Main window toolbar only when a Base Map window is open.	
	New folder	Insert a new folder under the selected item in the Repository window.	
	Variable	Used in the Variables tab of a Menu Item Properties window of a base map to identify a variable.	

Table 6. Icons used within the interface (continued)

Icon	Name	Description	Keyboard Shortcut
	Conditions	Used in a base map or driver map to indicate that conditions (filters or variables) are associated with a menu item.	
	Hidden system menu	Used in a base map to indicate a menu that can be accessed only by using a system-menu keyword.	

Module repository

During the ImageUltra Builder installation, the New Repository Wizard prompted you to define a location for your repository. In most cases, a single module repository is sufficient for storing all of your modules. The Repository window (redesigned for versions 2.0 and later) enables you to organize, categorize, and sort large numbers of modules. However, if you need additional repositories, you can create them after the initial installation of the ImageUltra Builder program. When you open the ImageUltra program, the workspace automatically associates itself with the last module repository used and opens the appropriate Repository window. The name of the active repository is shown on the title bar of the Repository window.

Note: If you upgraded from an earlier version of the ImageUltra Builder program, and you attempt to use a repository created by the earlier version, you will be prompted to upgrade the repository to the 2.2 level. Version 2.2 cannot use a repository from a previous version until the repository is updated. Once the repository is updated to the 2.2 level, the previous versions cannot use the updated repository. Therefore, it is a good idea to upgrade all of your ImageUltra Builder installations to the 2.2 level before updating the repositories.

Working with repositories

To make a new repository, do the following:

1. From the Main window menu bar, click **File**.
2. Click **New Repository....** The New Repository Wizard opens.
3. Click **Next**.
4. In the Location field, use the browse button to select the location where you want the new repository folder to reside. The new repository folder can reside on any disk to which the ImageUltra Builder console has access.
5. In the Server field, type the full name of the server on which you want to create the folder that will contain the repository. This must be a new folder name, not the name of an existing folder. This folder appends itself to the path specified in the Location field.

   Note: The folder name will not be used as the name for the new repository. You will have the opportunity to define a name for the repository later in this procedure.

   Note: You can also enter the IP address of the server in the Server field. For a local server, localhost can also be entered.

6. Click **Next**.

7. In the Username field, type your administrator username.

   Note: You must be an administrator to create new repositories.

8. In the Password field, type your administrator password.

9. Click **Next**.

10. In the Description field, type a name that you want to use to identify the repository.

11. Check the Archive Repository check box if you want to use this repository as a backup or as an archive repository.

12. Click **Next.**

13. Select a unique Repository ID for this repository.

    Note: This identifier must be different from any identifier used in an existing repository within your company. This identifier will be used in the file name of all modules created in this repository. Assigning unique identifiers to each repository prevents the possibility of overwriting an existing map or module when importing modules from a Smart Image that was deployed from a folder, or from another repository.

14. Click **Finish.**

To open a repository or change to a different repository, do the following:

1. From the Main window menu bar, click **File**.

2. Click **Open Repository...**. A list of recently accessed repositories is displayed.

3. Enter the server name for the server on which your repository resides, and then click **List**. This displays a list of repositories.

4. In the Username field, type your username.

5. In the Password field, type your password.

6. Select a repository.

7. Click **Open**.

To change to a module repository that has been recently accessed, do the following:

1. From the Main window menu bar, click **File**.

2. Click **Recent Repositories**.

3. Click the desired repository.

Exporting maps and modules

You can also export maps, modules, ImageUltra export packages, or the complete contents of a repository to a folder outside of a repository for archive purposes. This is very helpful if you want to clean out unused maps and modules from your repository. For example, if you want to archive to a network or CD, saving the folder structure, use the ImageUltra export package option.

Note: This package saves your maps, modules, and folder structure when it is imported by the Import Wizard.

Once you export a map or module to an outside folder, you can erase the map or module from your repository. However, if you ever need the map, module, or package in the future, you can import it back into the repository.

Note: While exported modules are in an external folder, they can no longer be modified, referenced, or used in any way as part of the image process.

The options for exporting are as follows:

- Export to an existing archive repository
- Export to an ImageUltra export package

Export to an existing archive repository

To begin the export process, do the following:

1. In the Export Wizard, click the appropriate radio button; then, click **Next**.
2. Create a repository by using the New Repository Wizard; and check the archive check-box that you want to use for archive purposes. This repository can be anywhere on the network where the ImageUltra Builder console has access.
3. Open the repository from which you want to export the module.
4. From the Main window menu bar, click **Tools** and then click **Export...**. The Export Wizard opens.
5. In the Export Wizard, click the appropriate radio button; then, click **Next**.
6. In the **Server** field, use the drop-down menu to select the desired repository.
7. In the **Username** field, type your username.
8. In the **Password** field, type your password.
9. Use the **List** button to display the list of archive repositories on that server.
10. Select a repository from the list where you want the exported maps and modules to reside, and click **Next**.

    Note: You can only export maps and modules to an existing archive repository. To create an archive repository, click the **CreateArchive** button, then click **List** to select the repository you want to export to.

11. A folder view of the repository is shown. If you want to work with a categorical view, click View by categories.
12. Expand the view if necessary; then, place a check mark in the check box for each map, module, folder, or category you want to export.

    Note: If you want the folder structure to be maintained at the external folder when you export the modules, make sure a check mark is in the Export folder structure check box. If you want all exported maps and modules to the root level of the external folder, remove the check mark.

13. Click **Next**.
14. Click **Export**.
15. When all selected maps and modules have been exported, click **Finish**

Archived modules can be added back into a repository using the Import Wizard. To use the Import Wizard, do the following:

1. Open the repository in which you want the imported module to reside.
2. From the Main window menu bar, click **Tools**; then click **Import...**. The Import Wizard opens.
3. Follow the instructions on the screen.

Export to an ImageUltra export package

To begin the export process, do the following:

1. In the Export Wizard, click the appropriate radio button; then, click **Next**.

   Note: If you want the folder structure to be maintained at the external folder when you export the modules, make sure a check mark is in the Export folder structure check box. If you want all exported maps and modules to the root level of the external folder, remove the check mark.

2. Create a repository using the New Repository Wizard, and check the archive check box to use for archive purposes. This repository can be anywhere on the network where the ImageUltra Builder console has access.

3. Open the repository from which you want to export the module.

4. From the Main window menu bar, click **Tools**; and then, click **Export....** The Export Wizard opens.

5. In the Export Wizard, click the **Export to an ImageUltra export package** radio button; then, click **Next**.

6. Type the path, or use the **Browse** button, to specify the directory from which you want to export maps and modules.

   Note: You can only export maps and modules to an existing archive repository. To create an archive repository, click the CreateArchive button, then click List to select the repository that you want to export to.

7. A folder view of the repository is shown. If you want to work with a categorical view, click **View by categories**.

8. Expand the view if necessary; then, place a check mark in the check box for each map, module, folder, or category you want to export.

9. Click **Next**.

10. Click **Export**.

11. When all selected maps and modules have been exported, click **Finish**.

The Repository window

The Repository window is the starting point for every operation in the ImageUltra Builder program. You use the Repository window to view the contents of the repository, add maps and modules to the repository, open and modify maps and modules contained within the repository, build modules, and create deployment media.

To help you manage and work with large numbers of maps and modules, the Repository window provides aids to help you organize, sort, categorize, and archive maps and modules. Also included is a search feature to help you locate a specific map or module, or to locate groups of maps and modules with common characteristics.

Notes:

1. For more assistance, the **View-->Source** option has been added to the Repository Window for ImageUltra Builder. Selecting the **View-->Source** menu item causes the highlighted module's Source Directory folder contents to appear in a Windows Explorer window, and allows you to browse a folder's contents without having to open the module.

2. The **Tools--> Set Password** option has also been added to the Repository Window for ImageUltra Builder, which allows you to change your password.

The Repository window is divided into two panes: the Folder area and the Contents area. The left pane is the Folder area. You can add as many folders and subfolders as you feel are necessary to organize your maps and modules. The right pane is the Contents area, where you view the contents of the repository or contents of any folder. Along the top of the Contents area are tabs that enable you to view the contents of the repository or any selected folder in a variety of ways:

- All maps and modules
- Operating-system modules only (base operating-system modules, add-on operating-system modules, partitioning modules, UNATTEND.TXT modules, ImageUltra Customization module, and so on)
- Application modules only
- Device-driver modules only
- Filter modules only
- Utility modules only
- Base maps only
- Driver maps only
- ImageUltra Builder 3.0 template modules

By default, ImageUltra Builder program provides one folder in the repository, the ImageUltra folder. The ImageUltra folder contains sample maps and ready-to-use utility modules, filter modules, special-purpose application modules, and special-purpose operating-system modules. Also included are module templates for creating I386 base operating-system modules and UNATTEND.TXT modules. The name of each map, module, and template provided by the ImageUltra Builder starts with *ImageUltra 2.0 -*, *ImageUltra 2.01 -*, or *ImageUltra 2.2 -* to identify the version of the ImageUltra Builder program for which these maps, modules, or templates were originally developed or updated.

Although the Repository window looks similar to Windows Explorer and some of the functionality is similar to Windows Explorer, it is important for you to understand that you are working with a database, not a file system. Each entry in the Repository window is a reference to a map or module, not the name of the map or module itself. By using references, you can have multiple entries for the same map or module in different folders, and when you modify the map or module in one folder, the same changes apply to all entries for that specific map or module in all folders.

Entries and icons

Each entry in the repository has an icon associated with it. Module icons serve the following purposes:

- They provide a visual indicator for the type of module (operating-system module, application module, device-driver module, filter module, or utility module).
- They distinguish a container module from a standard module.
- They indicate whether the module is in the test state or gold state. If the icon is gold, the module is in the gold state. When a module is in the gold state, the module is locked and you cannot make changes to it. If the icon is any other color but gold, the module is in the test state. When a module is in the test state, you can make changes to the module attributes and source files as often as is needed.

Map icons serve the following purposes:

- They provide a visual indicator for the type of map (base map or driver map).
- They indicate whether the map is in the test state or gold state. If the icon is gold, the module is in the gold state.

Because creating a module entry in the repository and building the module are two separate operations, the ImageUltra Builder interface provides a means by which you can tell whether or not a module has been built. When you look in the Repository window, if the text for module entry is shown in grey, the module has not been built; if the text for module entry is shown in black, the module has been built.

These icon and module entry conventions also apply to all other windows used in the ImageUltra Builder interface.

Manipulating entries

Moving and copying map and module entries in the Repository window is very similar to using Windows Explorer; you use the right mouse button to drag and drop the selected entry into the desired folder. Then, when you release the mouse button, you specify if the operation is a move operation or a copy operation. If you copy a map or module entry into a different folder, that entry is a reference to the same map or module as the original entry. If you copy an entry into the same folder as the original, you create a new map or module that has "Copy of" added to the beginning of the entry. Any modification to this new map or module does not affect the original, and any modification to the original map or module does not affect the new map or module. You can also copy multiple maps and modules in a single operation using the same techniques as used in Windows Explorer.

Deleting a map or module entry is also similar to using Windows Explorer; you right-click on the appropriate map or module entry, select Delete, and then confirm the delete operation. However, because you might have multiple instances of an entry in different folders, the actual map or module is not deleted from the repository until the last instance of an entry is deleted. Then, when the last instance of the entry is deleted, a prompt appears asking if you want to delete the map or module from the repository.

You can also use keystrokes instead of a mouse to achieve the same results. For example, to copy an entry into another folder you do the following:

1. While in the Repository window, repeatedly press the Tab key until you are located in the correct area of the window; then, use the Up Arrow or Down Arrow key to select the appropriate entry.
2. Press Alt+E to open the Edit menu, use the Down Arrow key to select **Copy**, and then press Enter. (Or, you can use the Ctrl+C shortcut).
3. Repeatedly press the Tab key until you are located in the same area of the window where the folder is located; then, use the Down Arrow key to select the folder into which you want to copy the entry.
4. Press Alt+E to open the Edit menu, use the Down Arrow key to select **Paste**, and then press Enter. (Or, you can use the Ctrl+V shortcut).

Searching for specific maps and modules

If you are working with large numbers of maps and modules, you might have to locate a specific map or module or a group of maps or modules that share a common characteristic.

The search engine enables you to perform searches using either free-form text or predefined search options. An advanced search feature enables you to search for maps and modules based on usage. For example, an advanced search can produce results that identify modules used by a specific map, all maps that use a specific module, or all modules that are not used by any map. In addition, you can customize the search so that it searches an entire repository or specific folders within a repository.

To perform a basic search for a map or module, do the following:
1. From the Main window menu bar, click **Tools** and then click **Search...**.
2. In the **Objects to search for** field, place a check mark in the check box for the type of map or module for which you want to search.
3. In the **Text to search for** field, type a string that is common to all modules you want to locate. This step is optional and is typically used to narrow down search results.
4. In the **Look in** field, use the drop-down menu to specify whether you want to search the entire repository or if you want to search within a specific folder contained in the repository.
5. Click the **Search now** button. The search results are displayed in the right-side of the search window.

For more advanced searches, click **Search Options >>** to display a list of search criteria and advanced search options. Use this list to help specify certain map or module characteristics that you want to include in your search. For example, the State search criteria enables you to search for only those maps or modules that have been promoted to the gold state.

For detailed information about using the search engine and performing advanced searches, see the ImageUltra Builder help system.

Map windows

When you double-click either a base map or driver map in the Repository window, a Map window opens. Map windows are divided into two panes: the Tree Structure pane and the Contents pane.

The left pane is the Tree Structure pane, which contains a root entry and menu-item entries. In a base map, menu-item entries control the flow and content of the menus displayed at the target computer during the installation process. In a driver map, menu-item entries are optional. When used in a driver map, menu-item entries enable a developer to annotate or label groups of device drivers or assign filters to one or more groups of device-driver modules.

The right pane is the Contents pane, which includes all of the modules associated with the menu-item entry selected in the Tree Structure pane.

Within the Map window, you can do the following:
- Delete a menu item from the tree structure
- Insert a menu item in the tree structure
- Insert a module in the tree structure
- Move a menu item up or down in the tree structure
- Assign filter and utilities to menu items in the tree structure of a base-map
- Assign filters to menu items in the tree structure of a driver map

Map Settings window

When a Map window is open, you can view or modify the settings associated with that map. To open the Map Settings window, do the following:

1. Open the desired map from the Repository window.
2. In the Main window menu bar, click **Tools**, and then click **Settings...**. (Or, you can right-click anywhere in the right pane, and select **Settings...**).

Base Map Appearance Under Windows PE

Under Windows PE, a base map appears within an actual windowing display. A menu item can appear in the form of one of the following familiar Windows controls:

- Static text
- Icon text
- Edit box or Prompt
- Checkbox
- Radio button

Below are examples of these controls:

- Static text: This control signifies unselectable text. It may be defined by creating a menu item, the text of which begins with an ampersand (&) character and which does not have a prompt variable associated with it. It does have a value defined for the Icon setting in its properties, as well.
- Icon text: This control also signifies unselectable text; and, like the Static text control, the Icon text control also has no prompt variable associated with it. It does have a value defined for the Icon setting in its properties, as well.
- Edit box: This is a control which prompts the user for input and has a prompt variable associated with it.
- Checkbox: This is a control which enables a user to choose multiple selections from a list for which checkbox has been selected in its properties, and whose text does not begin with an ampersand (&) character.
- Radio button: This control enables a user to make a single choice from a list in which all of the choices are mutually exclusive.

The user navigates through the menu by selecting items. That is, the next menu to appear in the interface is determined by which menu item the user selects. In ImageUltra Builder 3.0 it is possible to create a menu that consists solely of unselectable static text items. Also, in a multi-select menu, the user can select zero or more check boxes. To determine which menu should appear next in these situations, the following rules are enforced:

- For single-select lists (lists with radio buttons):

  Since one, and only one, radio button must be selected before the user can continue to the next menu, the selected radio button determines the next menu.

- for multi-select lists (lists with checkboxes):

  If one or more checkboxes are selected, the last selected checkbox determines the next menu. If no checkbox is selected, the last selected checkbox determines the next menu to appear as well.

- for lists consisting solely of unselectable static text items:

  The last static text item determines the next menu.

Menu item text is automatically flowed (i.e. word-wrapped) to fit within the console area of the map viewer window. The text associated with a menu item may contain certain HTML elements. The following elements are recognized:

- **<B></B>**

  Causes the text within the elements to be **boldfaced**;

- **<I></I>**

  Causes the text within the elements to be *italicized*;

- **<U></U>**

  Causes the text within the elements to be underlined;

- **
**

  Causes a line break within the text;

- **<NOBR>**

  Turns off normal word-wrapping for the text within these elements;

- **ACCEL**

  Immediately precedes the character which represents the mnemonic for the menu item. A mnemonic, sometimes called a keyboard accelerator, is a character which, when typed simultaneously with the ALT key depressed, results in focus being set to the control represented by the menu item. (If the item is a checkbox or radio button, it is also selected).

Base-map settings

In the Map Settings window for a base map, you can do the following:

- From the General tab, you can view the map name and add comments about the map. The Comments field supports up to 1024 characters. You can also assign keywords, which are used in conjunction with deployment media (*ImageUltra Builder 3.0 Deployment CDs*) to minimize the number of maps that can be chosen during the deployment phase. If keywords are assigned to the deployment media, only those maps that have matching keywords can be selected during the deployment phase.

  The System menu keyword: Use this field to assign a keyword to a hidden system menu. If this keyword is typed anytime while the installation menus are visible, a hidden system menu is displayed. The content of the hidden system menu is defined as part of the base map. For example, you might want a set of utilities available during the deployment process. By typing the keyword, you could access the utilities as part of a hidden system menu anytime during deployment. See Chapter 6, "Map construction overview," on page 79 for details about creating a hidden system menu.

- From the User Information tab, you can define either end-user settings to be used during the installation or prompts for end-user settings that appear during the installation.

- From the Before/After Menu Options tab, you can:
  - Specify one or more utilities to run at the target computer before the installation-menu system is displayed
  - Specify one or more utilities to run at the target computer immediately after making the final selection from the last installation menu.

- From the Install Sequence window, you can modify the order in which modules are installed on the target computer. Modifications are limited to defining the order of operating-system modules (for installation during the audit boot) and modules that are installed during the customer first boot. Customer-first-boot modules have an implied installation order based on *install slots* defined by the

module attributes. The Install Sequence window enables you to refine the order for modules that share a common install slot.

Within the Install Sequence window, there are three categories: First, No preference, and Last. By moving modules within these three categories, you determine the order of which modules get installed first and which modules get installed last. If order is not important, you leave the modules in the No preference category. For additional information about setting the installation sequence, see Chapter 9, "Specifying the module-installation order," on page 167.

...... The order in which customer-first-boot modules or base operating system modules are installed. For more information on how to define the order in which either customer-first-boot modules or base operating-system modules are installed, refer to the ImageUltra help system for details.

- From the Image Cleanup tab, you can specify which action to take on the Smart Image after the installation is complete:
  - **Delete none** keeps the entire Smart Image intact in the service partition of the target computer.
  - **Delete unused** deletes all unused modules from the Smart Image, keeps only those modules required to recreate the installed image in the service partition, and resizes the service partition to accommodate those modules.
  - **Delete all** deletes the entire Smart Image and deletes the service partition if the service partition was used as the staging area.
- From the Network tab, you can specify whether or not to perform a network-sync operation and select the type of deployment/installation method to use

  To enable the network-sync function, place a check mark in the **Use Network Sync** check box. When enabled, the network-sync function looks for the latest version of this map during deployment and during installation.

  To select the type of deployment/installation, use one of the three following radio buttons provided in the **Direct-network install** area:
  - **Yes:** You select this radio button if you want to perform a direct-network installation, where deployment and installation are done over the network.
  - **No:** You select this radio button if you want to perform a standard network deployment with local installation, or if you want to use CDs for deployment and installation.
  - **Prompt the user when the image is deployed:** You select this radio button if you want the person at the target computer to determine which type of deployment/installation to perform.
- From the Menu Appearance tab, you can control the display of menu items within the map viewer window. These settings, which only apply to base maps, can be defined at both the map level (global) and the menu-item (local) level. Global values are set within the Map Appearance tab of the Map Settings dialog. Local overrides are defined within the Menu Appearance and Background Image tabs of the Menu Item Properties window.

  If a setting is defined in the Menu Appearance tab of the Map Settings dialog, that setting becomes the default value for all menu items within the base map which do not themselves define that setting. If a menu item defines the setting, then that value overrides the global setting.
  - **Margins:** This setting specifies the size of the left and right margins for items within a menu. Values can be specified in units of pixels or as a percentage of the menu viewer window width.
  - **Colors:** Colors for the following aspects (Items) of menus can be customized:
    - Banner text

- Banner background
- Menu text
- Menu background
- Edit box background

– Show banner:

Determines whether banners for menus can be displayed. By default, each menu displays a banner along the top of the map viewer window; however, a menu item can hide the banner for the display of its own submenu items.

– Deploy-time logo path:

The path of an image. Windows bitmap (.bmp) file, which will be displayed right-justified within each banner. The path must exist at deploy time.

– Background Image:

This feature enables the display of a background image within menus.

– Deploy-time image path:

The path of an image. Windows bitmap (.bmp) file, which will be displayed as a background image within the menu.

– Alignment:

This feature specifies the alignment, or position, of the background image within the menu. Values are the following:

- Top Left
- Top Center
- Top Right
- Center Left
- Center Center
- Center Right
- Bottom Left
- Bottom Center
- Bottom right

– Attachment:

This feature determines whether the background image is fixed with regard to the menu contents or if it scrolls with the menu contents (assuming scroll bars are displayed). Available values for this feature are Fixed and Scroll.

– Previous menu key: Use this setting to define the key to be used to return to a previous menu. The default setting for this key is Esc.

– Exit menu key: Use this setting to define a key that can be used to exit from the installation-menu system and get to a command-prompt screen. The default setting for the exit key is F3. To disable the exit-menu function, leave this setting blank.

- From the new *Menu Window* tab, you may change settings which apply to the deploy-time map viewer window (formerly DOE). These settings only apply to base maps.

- From the Partition tab, you can define a custom size for the service partition. A custom service-partition size enables you to leave room for other products that share the service partition.

You can set a custom partition size using any of the following methods:

– **A fixed size:** To set a fixed size, click the **A fixed size** radio button; then, type the size of the partition (in megabytes) in the space provided.

– **A percentage of the drive:** To set a partition size based on a percentage of the drive, click the **A percentage of the drive** radio button; then, use the slider bar to select a percentage.

– **Based on size of image:** To set a partition size with a fixed-size buffer, click the **Based on size of image** radio button. Then, put a check mark in the **Add an additional cushion** check box and type the buffer size in the space provided. To set a maximum partition size, place a check mark in the **Partition size may not exceed** check box and use the slider bar to select a percentage of the drive.

From the Install Sequence window, you can modify the order in which modules are installed on the target computer. Modifications are limited to defining the order of operating-system modules (for installation during the audit boot) and modules that are installed during the customer first boot. Customer-first-boot modules have an implied installation order based on *install slots* defined by the module attributes. The Install Order tab enables you to refine the order for modules that share a common install slot.

Within the Install Sequence window, there are three categories:

- First
- No preference
- Last

By moving modules within these three categories, you determine the order of which modules get installed first and which modules get installed last. If order is not important, you leave the modules in the No preference category. For additional information about setting the installation sequence, see Chapter 9, "Specifying the module-installation order," on page 167.

Driver-map settings

In the Map Settings window for a driver map, you can do the following:

- From the General tab, you can view the map name and add comments about the map. The Comments field supports up to 1024 characters. You can also assign keywords, which are used in conjunction with deployment media (*ImageUltra Builder Deployment Discs* and *ImageUltra Builder Deployment CDs*) to minimize the number of maps that can be chosen during the deployment phase. If keywords are assigned to the deployment media, only those maps that have matching keywords can be selected during the deployment phase.
- From the Network tab, you can:
  - Specify whether or not to perform a network-sync operation during deployment and installation to look for the latest version of this driver map.
  - Enable or disable driver-map filter processing. When enabled, only those device-driver modules that apply to the target-computer machine type (based on the Model Check filter provided with the ImageUltra Builder program) are deployed to the target computer. When disabled, all device-driver modules defined in the map are deployed to the target computer, but only those that are needed get installed.

Menu Item Properties window

In the Menu Items Properties window for a base map, you can do the following:

- From the General tab, you can view or change the description of the root entry or any menu-item entry. When you change the root entry description, the base-map entry in the repository also changes everywhere where that base map

is referenced. Also, from this tab you can control whether the menu item is left justified, right justified, or centered when viewed at the target computer.
- From the Submenu Items tab, you can do the following:
  – In the Title field, define a title or replace the title with an instruction associated with the menu choice. This text will appear within the banner, if displayed, for the child menu items of this menu item. The text may contain HTML markup.
  – In the Allow Multiple Selections field, create a multi-select list from the child menu items. A multi-select list is a list in which each child menu item appears as a checkbox control. The following settings affect the selection among these checkboxes:
    - Minimum: Specifies the minimum number of checkboxes which must be selected before the user may continue to the next page.
    - Maximum: Specifies a maximum number of checkboxes which the user may select.
  – In the Show Banner field, you control whether a banner is displayed for the child menu items of this menu item.
  – In the Deploy-time logo path, you provide the path to the image which will be displayed right-justified within the banner.
  – In the Automatically Select Single Item check box, define whether or not you want the installation program to automatically make a selection when only one menu item is present.
  – In the Allow Multiple Selections check box, define whether or not a user can make multiple selections from this menu. Provide an acceptable range by listing values in both the minimum and maximum fields.
- From the Utility tab (menu items only), you can associate a utility with a menu item, input any parameters associated with the utility to get the desired result, and determine when the utility is to run (immediately when the menu item is selected, or placed in a queue to be run after the installation menus are finished).
- From the Variables tab, you can control the behavior of each individual menu item, except for prompt variable, which is now defined using the User Input tab, which only appears for base maps. For example, you can hard code a variable for a menu item or you can define a prompt so the person controlling the deployment at the target computer can provide a value that gets assigned to a variable. See Chapter 6, "Map construction overview," on page 79 for more information about the use of variables.
- User Input tab: This tab contains settings for the menu item's prompt variable, and it only appears for base maps:
  – Width...Specifies the width of the edit box control in units equal to the average character width of the font in use.
  – Max. characters...Specifies the maximum number of characters which the user can type within the edit box.
  – Allow only...If checked, allows only the following types of characters to be typed in the edit box:
    - Alphabet...Alphabetic characters (a through z and A through Z)
    - Digits...Numbers 0 through 9
    - Blanks...The space character
    - Other...Those characters specified
  – Prevent only...If checked, suppresses the following types of characters from being typed in the edit box:

- Alphabet...Alphabetic characters (a through z and A through Z)
- Digits...Numbers 0 through 9
- Blanks...The space character
- Other...Those characters specified

- Menu Appearance tab
- Background Image tab: This tab provides choices that enable you to customize how you want to display the background image within the menu that is seen on the target computer during the installation process. This tab only appears for base maps and menu items listed within the map. The following is a listing of the fields on the Background Image tab:
  - Deploy-time logo path: Enter the path of an image, Windows bitmap file, that will display as a background image within each menu item.
  - Alignment settings: This field enables you to customize the alignment or position of the background image within the menu item. The Alignment drop-down menu enables you to select from the following values:
    - Top Left
    - Top Center
    - Top Right
    - Center Left
    - CenterCenter
    - Center Right
    - Bottom Left
    - Bottom Center
    - Bottom Right

    Attachment setting: This field enables you to select the value related to how the background image is positioned within the menu contents. The values are Fixed and Scroll.

In the Menu Items Properties window for a driver map, you have the following tabs:

- From the General tab, you can view the or change the description of the root entry or any menu-item entry. When you change root entry description, the driver-map entry in the repository also changes everywhere where that driver map is referenced.
- From the Conditions tab, you associate filters with the menu-item entry. The use of filters is optional, but helpful in determining which modules are appropriate for certain target computers.
- From the Exclude tab, you can select modules from your driver map that are to be excluded from the installation process if that menu item is selected.

Menu Link Properties window

This window contains the following tab to use when defining general and specific properties for menu links.

- General

Module windows

When you double-click a module entry in the repository, a module window opens for the selected module. Within the module window, you can view or define certain attributes, such as the language, location of the source files, associated filters, and module-installation behavior. You also can include free-form text comments about the module, up to 1024 characters. The attributes that you can view or modify vary depending on the type of module you select. The five types of module windows are as follows:

- Operating System window
- Application window
- Device Driver window
- Filter window
- Utility window

The attributes define key information about an associated module that affect content, behavior, and installation. The attributes must be filled in completely before you build a module. If a module is imported from a source computer, all required attributes are already filled in. However, it is a good idea to open the associated module window and review the attribute values to ensure that they are suitable for your specific implementation.

When you build a module from scratch, you use the New Module Wizard to create the entry in the repository and fill in the attributes. You must complete the associated attributes before you can build the module.

Modifying module attributes and map settings

By default, when you import or create a map or module, it is placed in the repository in the test state. As long as a module is in the test state, you can modify its attributes and source files and rebuild the module as many times as is needed. As long as a map is in the test state, you can modify its tree structure and settings. Once a map or module has been promoted to the gold state, it is locked and you can no longer make changes to it.

It is imperative that you test Smart Images while the maps and modules are in the test state. Once you are comfortable that all of the maps and modules are working correctly, *promote* them to the gold state to ensure that they are not accidentally modified or overwritten in the future.

You can determine whether a map or module is in the test state or gold state by using either of the following methods:

Method 1: Icon identification

To determine whether a module is in the test state or gold state using the icon identification method, do the following:

1. Open the appropriate Repository window.
2. In the left pane, navigate to the folder that contains the desired map or module entry and open that folder.
3. In the right pane, locate the desired map or module.
   - If the icon for the module is any color other than gold, the module is in the test state.

- If the icon for the module is gold in color, the module is in the gold state.

Method 2: Properties identification

This method is useful if you have difficulty distinguishing colors. To determine whether a module is in the test state or golden state using the properties-identification method, do the following:

1. Open the appropriate Repository window.
2. In the left pane, navigate to the folder that contains the desired map or module entry and open that folder.
3. In the right pane, right-click the appropriate map or module and then select Properties. If the Gold checkbox located near the bottom of the tab is marked, the map or module is in the gold state. If the Gold checkbox is not marked, the map or module is in the test state.

Tools

The ImageUltra Builder program makes several tools available from the menu bar to assist with the image building process and the management of maps and modules.

Repository tools

The following tools are available from the Tools entry in the Main window menu bar when the Repository window is the only window open.

Note: These tools are not available when a Driver Map window or Base Map window is open. You must close the Driver Map window or Base Map window to access the tools described in this section. For information about the tools associated with base maps and driver maps, see "Map tools" on page 76.

- **Build Module**

  The Build Module selection enables you to build a module from source files. A module must be highlighted in the Repository window before this selection can be made.

- **Promote**

  The Promote selection enables you to promote a map or module from the test state to the gold state. A map or module must be highlighted in the Repository window before this selection can be made.

- **Import**

  The Import selection opens the Import Wizard. The Import Wizard prompts you through the steps required to import maps and modules from a source computer, CDs (*ThinkCentre and ThinkPad Recovery CDs* or *ImageUltra Builder Distribution CDs*), folders, an ImageUltra export package, and other repositories. A Repository window must be open before this selection can be made.

- **Export**

  The Export selection opens the Export Wizard. The Export Wizard prompts you through the steps required to archive one or more maps and modules to a folder outside of the repository. Exporting maps and modules to a folder is useful if you want to clean out unused maps and modules from the repository. You can also export packages that contain maps and modules with a unique structure. If you want to archive to a network or CD while saving the folder structure, use the export to an ImageUltra export package function.

Once a map or module has been exported, you can delete the original map or module from the repository. If you ever need the exported maps, modules, or packages in the future, you can use the Import Wizard to add them back into your repository.

- **Deploy**

  The Deploy selection opens the Deploy Wizard. The Deploy Wizard prompts you through the steps required to create a *Network Deployment CD*, *Network Deployment CD*, or stand-alone *Distribution CDs*. The deployment CD establishes a connection to the repository, provide an interface for selecting the desired maps, and deploy the appropriate maps and modules to the appropriate staging area (service partition, HPA, or network-staging folder). The standalone *Distribution CDs* contain all of the required maps and modules, and deploy them to the target computer without using a network. A Repository window must be open before this selection can be made.

- New Repository Wizard

  The New Repository Wizard guides you through the process of creating new repositories. To start the wizard, click File in the Main window menu bar; then, click New Repository.

- Migrate Repository

  The New Migrate Repository Wizard guides you through the process of migrating from a previous version of an ImageUltra repository. To start the wizard, click File in the Main window menu bar; then, click Migrate.

- **Network-Sync Settings**

  The Network-Sync Settings selection enables you to define replacement maps for maps that have already been deployed. A Repository window must be open before this selection can be made.

- **Get ImageUltra Customization Program**

  This selection provides instructions on how to locate and copy the ImageUltra Customization Program (IUBCP.EXE) to a CD or shared drive. The ImageUltra Customization Program is required when building the source for Portable-Sysprep Images. A Repository window must be open before this selection can be made.

- ImageUltra Administrator

  The new ImageUltra Administrator tool guides administrators through the process of managing servers and users, and users and their access to repositories. The Administrator tool secures repositories based on predetermined access levels set by the administrator. To start the tool, click Tools in the Main window menu bar; then, click Administration.

- Set Password

  The Set Password tool enables users to update or change their password. To start the tool, click Tools in the Main window menu bar; then, click Set Password.

Map tools

The following tools are available from the Tools entry in the Main window menu bar when a map window is open.

- **Settings**

  The Settings selection opens the Map Settings window for the Base Map window or Driver Map window that is currently active.

- **Preview**

The Preview selection enables you to preview the installation menus for the Base Map window that is currently active.

- **Map report**

  The Map report selection enables you to view a dynamically generated report that contains important information about the map and the installation process associated with the map.

Chapter 6. Map construction overview

Creating a fully functional map involves four fundamental steps:

- Creating a map entry in the module repository
- Populating the tree structure with modules

This chapter focuses on the concepts behind creating a map tree structure and defining certain settings that affect map behavior. For step-by-step instructions on creating a map entry, defining settings for a map, or any other process described in this chapter, refer to the ImageUltra Builder help system.

Creating a base-map tree structure

The New Map Wizard enables you to create a new base map from scratch or to create a new base map using an existing base map as a template. The New Map Wizard creates the new map entry in the repository, assigns the map a file name, and places the map in the correct location of the repository. To access the New Map Wizard for base maps, click **Insert** in the Main window menu bar, and then click **Base Map**.

If you create a new map based on an existing map, you simply make the necessary changes to the tree structure and attributes. If you create a new map from scratch, you are responsible for:

- Inserting *all* menu items
- Inserting *all* modules
- Defining *all* menu-item attributes

Inserting menu items in a base map

Menu items are responsible for the menu choices displayed at the target computer during the installation process. Menu items also determine the hierarchy within the base map, meaning that some menu items are subordinate to other menu items. Each menu item listed in a base map directly corresponds to a choice made at the target computer during the installation process. When a menu item is chosen during the installation process, that action directs the installation program to accumulate modules associated with that menu item or to display a new menu of subordinate menu items. In some cases, both actions occur simultaneously. When the person performing the installation makes the final selection from the final menu, the installation program installs all of the accumulated modules. The accumulation of menu choices during the installation process transforms the Smart Image into a customized image for a specific target computer.

As you develop your tree structure, you might want to think of the menu items as an outline for the installation process associated with the entire Smart Image.

Assume that you want to construct a base map that gives the installer a choice of installing Windows XP Professional or Windows 2000. The following illustration shows a base-map tree structure with two menu items (Windows XP Professional and Windows 2000) before modules or submenu items are added. Also shown is the corresponding default installation screen as it would appear at the target computer.
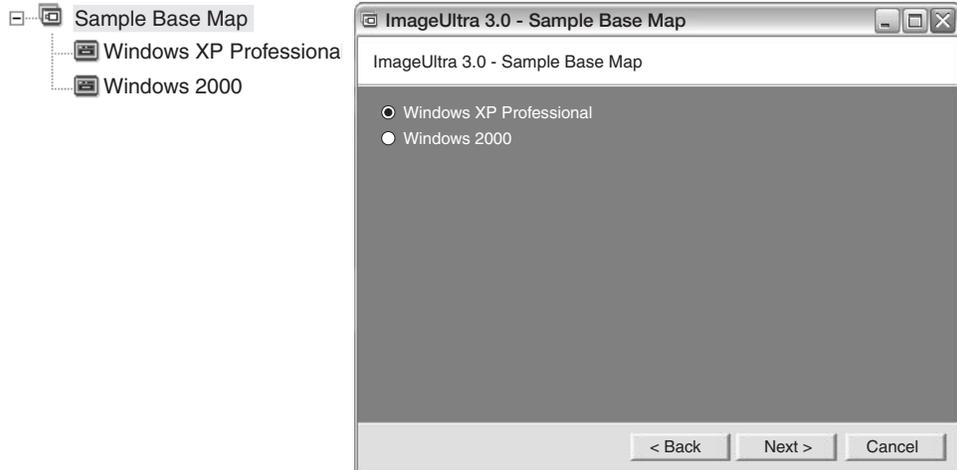
Notice how the menu items in the base map directly correlate to the menu items on the installation screen at the target computer. Optionally, you can use the Submenu Items tab in the Menu Items Properties window to override the default title and replace it with an instruction to clarify the action required at the target computer. For example, you might want to replace the default title with "Select an operating system" as shown in the following illustration.
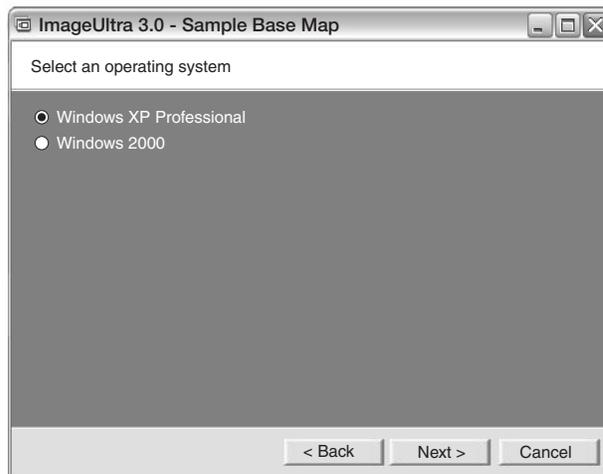

In this example, to change the default title "Sample Base Map" to "Select an operating system," you would take the following steps:

1. In the Repository window, double-click the appropriate base map entry. The Base Map window opens.
2. In the left pane of the Base Map window, highlight the root entry of the tree structure.
3. In the Main window menu bar, click **View** then click **Properties**. The Menu Item Properties window opens.
4. In the Menu Item Properties window, click the **Submenu Items** tab.
5. In the Title field, type **Select an operating system**.
6. In the Main window toolbar, click the **Save** icon and then close the Menu Item Properties window.

Now take the map one step further and give the installer language choices for the Windows XP Professional operating system by adding more subordinate menu items to the base map.

The following illustration shows the tree structure with two subordinate menu items (French and English) under the Windows XP Professional menu item. Also shown is the corresponding installation screen that the installer will see at the target computer after selecting Windows XP Professional from the first menu.
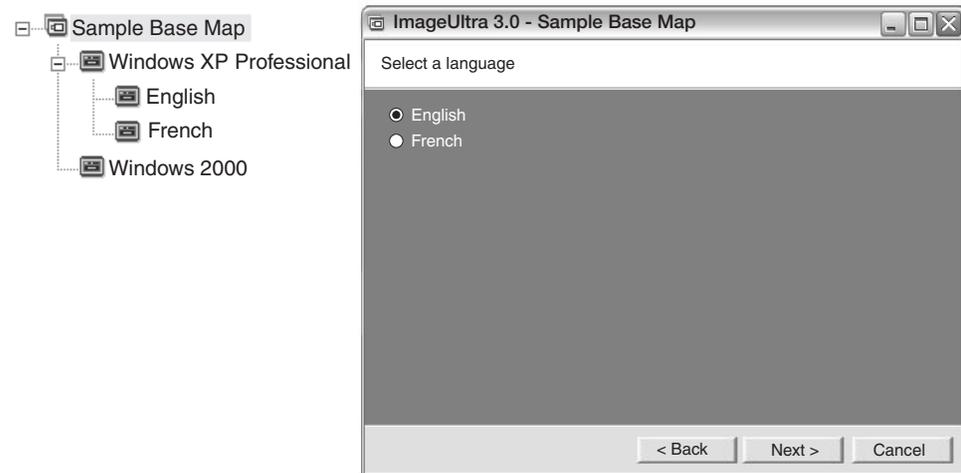
Notice how the menu items in the base map directly correlate to menu items on the installation screen at the target computer, but the title contains an instruction instead of the default title. To make the change shown in this example, you would do the following:

1. In the left pane of the Base Map window, highlight the Windows XP Professional entry in the tree structure.
2. In the Main window menu bar, click **View** then click **Properties**. The Menu Item Properties window opens.
3. In the Menu Item Properties window, click the **Submenu Items** tab.
4. In the Title field, type **Select a language**.
5. In the Main window toolbar, click the **Save** icon and then close the Menu Item Properties window.

You can continue building on this concept and add as many menu items and branches as needed to obtain a menu structure that gives you the desired result. You can also view a simulation of your menus anytime during the development process using the preview tool. See "Previewing your maps" on page 106 for details.

Inserting modules in a base map

Once you have established the hierarchy of a base map with menu items, you can start populating the map with operating-system modules and application modules.

One rule to remember is that at least one operating-system module must be associated with each branch that extends from the root node of a map tree. This can be accomplished using one of two methods:

- Branch method: At least one operating-system module must be nested somewhere within a branch (most common implementation). This method determines which operating system to install based on menu choices.
- Root method: If the same operating system will be used for each image defined in your map, you can insert the operating-system at the root node (least common implementation). This method automatically associates the operating system with every branch in your map.

Depending on the type of operating-system module associated with a given branch, you might have limitations as to whether you can associate application modules with that branch:

- If the operating-system module is a Hardware-Specific Image, you cannot associate application modules with that branch. If you attempt to associate application modules with that branch, the application modules will be ignored during the installation process. Only those applications contained within the Hardware-Specific Image itself will be installed.
- If the operating system module is an Ultra-Portable Image or Portable-Sysprep Image, you can associate application modules with that branch.

You insert modules in strategic locations in the base map. Module names never appear on the installation screen at the target computer. Typically, modules are associated with specific menu items so the installation of the modules is contingent upon the selection of a given menu item. As the person performing the installation progresses through the menu structure, the modules to be installed accumulate until the final selection is made from the final menu, at which time, all of the accumulated modules are installed.

Using the map that was used as an example earlier, the appropriate operating-system modules would be inserted in the right pane for the French and English menu items as shown in the following illustrations:
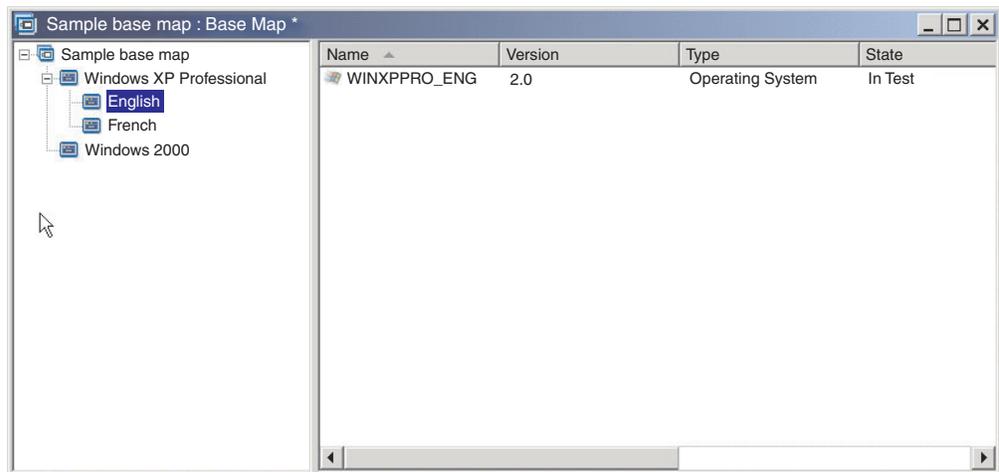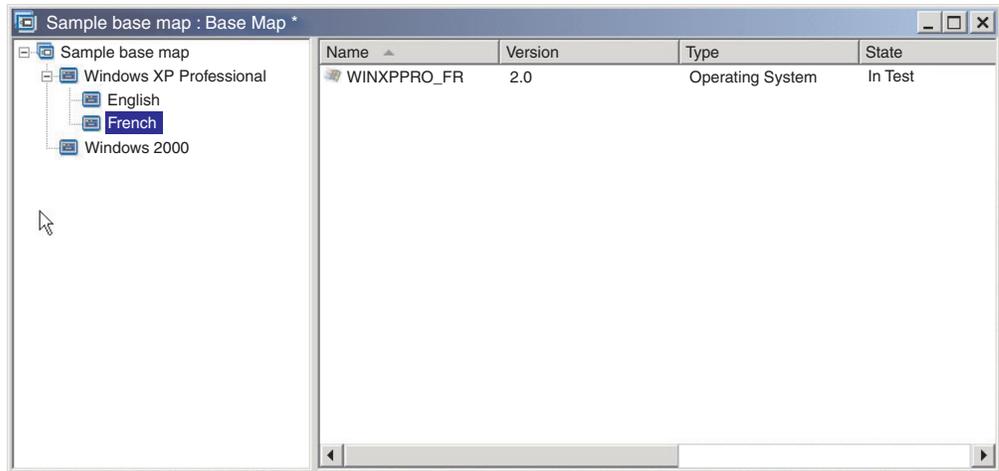
If the operating-system modules are either Ultra-Portable Images or
Portable-Sysprep Images, you can also insert application modules in the right pane
as shown in the following illustrations.
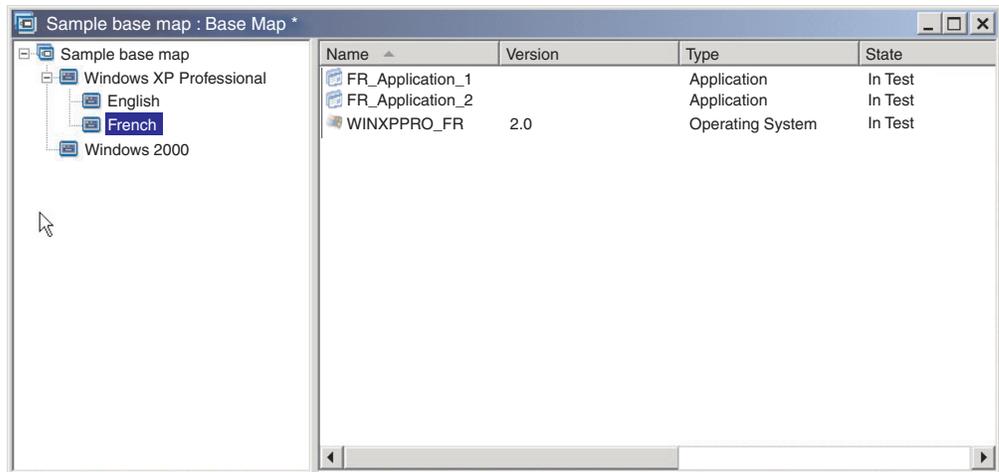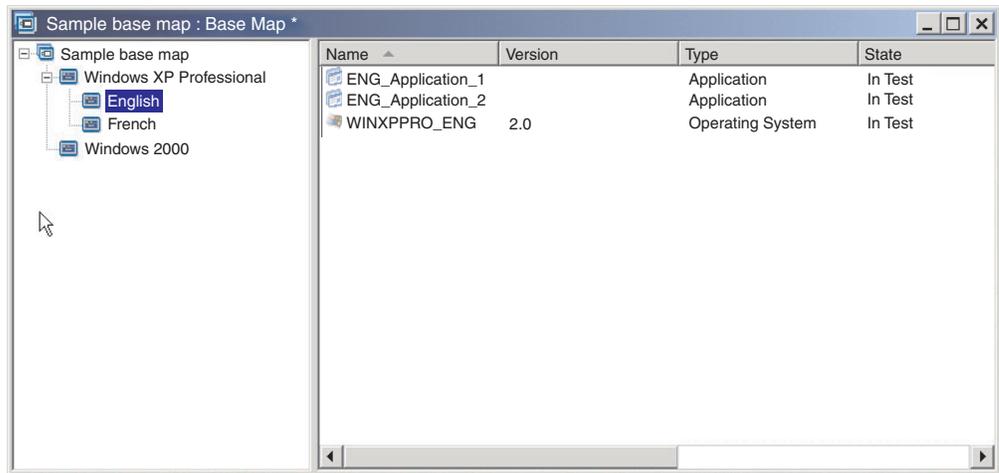
There is one important exception to associating modules with menu items. If you want to install one or more modules that are global in nature (that is, modules to be installed on every target computer using this map regardless of what menu items are selected), you can insert those modules in the right pane for the root entry of the map as shown in the following illustration.

Sample base map : Base Map *			
Name ▲	Version	Type	State
Windows XP Profe...		Menu Item	
Windows 2000		Menu Item	
Delete all user partitions		Operating System	
Global_Application_1		Application	In Test
Global_Application_2		Application	In Test
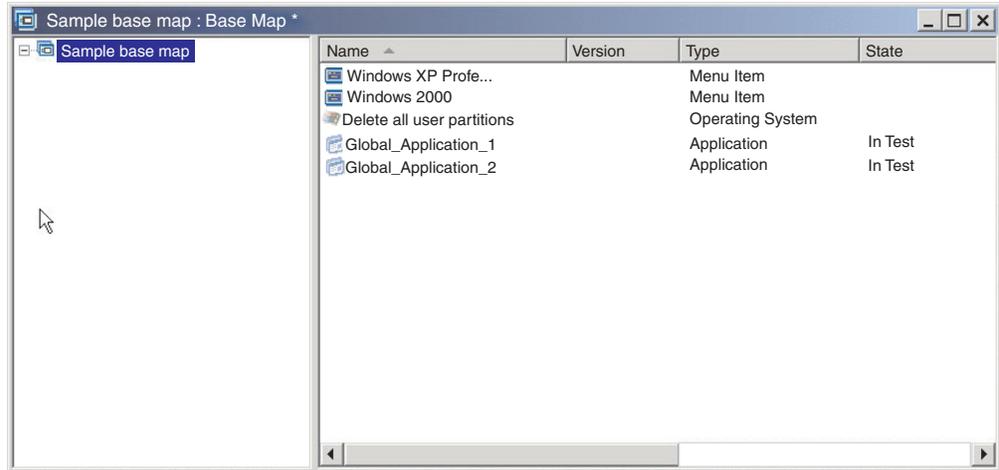
To insert a module in a base map, you do the following:

1. In the left pane of the Base Map window, highlight the root entry or the menu entry with which you want to associate the module.
2. In the Main window menu bar, click **Insert** then click the type of module you want to insert (**Operating System...** or **Application...**). A window opens listing all of the modules for that module type.
3. Select one or more of modules and then click **OK**.

Special considerations for I386 base operating-system modules

As discussed in earlier chapters, the I386 base operating-system module is the core element of an Ultra-Portable Image that can be deployed and installed on IBM branded and non-IBM branded computers, regardless of hardware configurations. However, whenever a map contains a I386 base operating-system module, that map also must contain the following modules:

- An appropriate Primary NTFS Partition partitioning module for the operating system being installed
- The ImageUltra Customization module
- The SysPrep module
- An UNATTEND.TXT module for the operating system and language being installed

These modules are designed to be used in conjunction with I386 base operating-system modules only. Do not attempt to use these modules in conjunction with any other type of base operating-system module.

Primary NTFS Partition partitioning module

The ImageUltra Builder program provides three variations of the Primary NTFS Partition partitioning modules:

- WinXP Pro Primary Partition

- WinXP Home Primary Partition
- Win2000 Primary Partition

Primary NTFS Partition partitioning modules are responsible for creating the primary partition, formatting the primary partition, and setting the primary partition as bootable in preparation for the operating system installation. These modules are provided as part of the ImageUltra Builder program and are ready to use without modification.

Because each of these partitioning modules is unique to a specific operating system, but has no dependency on language, insert the appropriate Primary NTFS Partition partitioning module under the menu item used to select the operating system. If your map is designed to install only one type of operating system (Windows XP Home, Windows XP Professional, or Windows 2000) and all installations are done using I386 base operating-system modules, you can insert the appropriate Primary NTFS Partition partitioning module under the root entry.

ImageUltra Customizations module

The ImageUltra Customizations module contains the ImageUltra Customization program, which provides the hooks required for the Ultra-Portable Images to make use of applications and device drivers defined in maps. The ImageUltra Customizations module is provided as part of the ImageUltra Builder program and is ready to use without modification.

Because the ImageUltra Customizations module is operating-system independent and language independent, you have a lot of latitude as to where you can insert this module. If your map contains multiple operating systems and all of the operating systems will be installed from I386 base operating-system modules, you can insert the ImageUltra Customizations module under the root entry. If your map also contains base operating-system modules for Portable-Sysprep Images or Hardware-Specific Images or imported ImageUltra-developed base operating-system modules, you must insert the ImageUltra Customizations module under the menu item used to select the operating system that will use the I386 base operating-system module. Do not attempt to use the ImageUltra Customizations module with Portable-Sysprep Images or an Ultra-Portable Images developed from imported ImageUltra-developed base operating-system modules.

UNATTEND.TXT module

The UNATTEND.TXT module contains the UNATTEND.TXT file that defines the values to be used during an unattended installation. The ImageUltra Builder program provides two sample UNATTEND.TXT files:
- UNATTEND.W2K: For use with Windows 2000
- UNATTEND.WXP: For use with Windows XP Professional and Windows XP Home

These sample files require some modification before they can be built as modules. Information about modifying these files is in Chapter 8, "Preparing source files for modules," on page 135.

After these files have been built as UNATTEND.TXT modules, you can insert them in your map. Because these modules are both operating system unique and language unique, in most cases you will probably insert the appropriate UNATTEND.TXT module under the menu item used to select the language of the operating system. If you are developing images for multiple customers, the

UNATTEND.TXT file typically has customer-unique information such as the company name and the PID (ProductKey or ProductID) associated with the license.

Using a container for I386 base operating-system modules and related modules

Some developers might find it easier to package the I386 base operating-system module, the appropriate Primary NTFS Partition partitioning module, the Sysprep module, the ImageUltra Customizations module, and the appropriate UNATTEND.TXT module into an operating-system container module. This approach enables you to insert a single container module in your map instead of having to insert individual modules. In most cases the container is both operating-system unique and language unique, so the container is typically inserted under the menu entry used to select the language for the operating system.

Compatibility with imported ImageUltra-developed application modules

The ImageUltra Customizations module provided as part of the ImageUltra Builder program is different from similarly named modules that can be imported from source IBM branded computers in that it does not contain the Sysprep function. Some imported ImageUltra-developed application modules might have a dependency on the Sysprep function and might not install correctly when used in images containing an I386 base operating-system module and the ImageUltra Customizations module. If an imported ImageUltra-developed application module does not install correctly, recreate the application module using the ImageUltra Builder program.

Using utilities in a base map

The use of utilities is optional. Utilities are Win32 applications that run before any part of an image is installed on a target computer. For example, you might want to run CHKDSK to determine the current condition of a target computer hard disk or PC-Doctor to determine the overall condition of a target computer before installing the actual image. One important difference between a utility and an application module is that a utility resides in the service partition only and never gets installed on any other active partition.

You can assign utilities to run in any of the following ways:
- Associate the utility with the map and have it run automatically at the beginning of the installation procedure, before any installation menus are displayed.
- Associate a utility with a menu item and have it run as soon as the menu item is selected.
- Associate a utility with a menu item and have it placed in a queue to run immediately after all of the installation menus have been completed, but before any modules get installed.
- Associate the utility with the map and have it run automatically after all of the installation menus have been completed, but before any modules get installed.

Assigning utilities to base-map menu items

Utilities are assigned to menu items in the tree structure of a base map. Each menu item can have a maximum of one utility associated with it. There are two approaches to assigning utilities to menu items:

- **Approach A**: Assign utilities to menu items that already contain modules. In this approach, the utility is automatically selected when the installer selects the existing menu item. It is transparent to the installer that a utility is being selected.

The following is a sample tree structure with utilities associated with the "Windows XP Professional" and "Windows 2000" menu items. If you compare this tree structure to the sample tree structure used earlier, you will notice that the tree structure itself does not change, but the icon associated with the menu item does change to show that a utility is associated with it. Notice the small hammer that is now part of the icons for the "Windows XP Professional" and "Windows 2000" menu items.
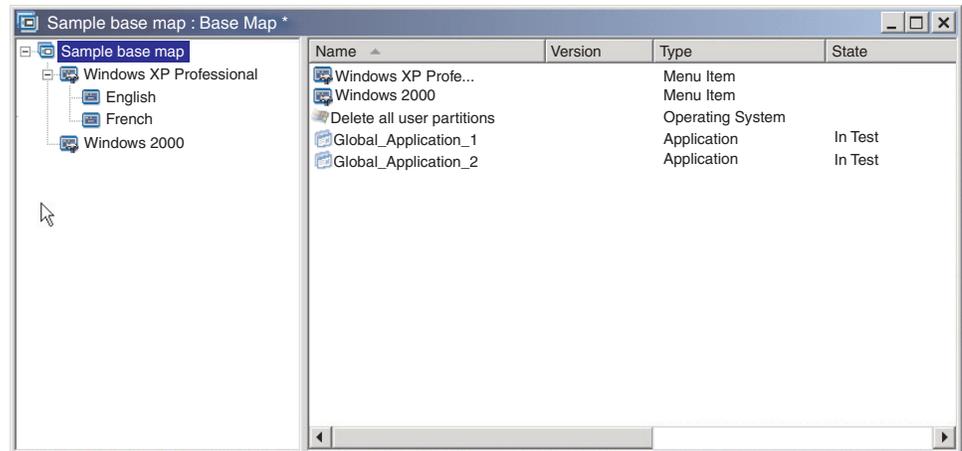
When using Approach A to assign a utility to a menu item, you do the following:

1. In the left pane of the Base Map window, select the menu item to which you want to add the utility.
2. In the Main window menu bar, click **View**, and then click **Properties**. The Menu Item Properties window opens.
3. Click the **Utility** tab.
4. In the Name field, use the drop-down menu to select a utility module from the repository.
5. In the Parameters field, type any parameters required to run the utility.
6. In the Run field, select when the utility is to run:
   - **Immediately:** The utility is run immediately when the menu item is selected at the target computer. This is the default setting.
   - **Queue:** The utility is queued to run after the final selection is made from the final menu in the sequence. All utilities that are queued are run before any files get installed.
7. Click the **Save** icon located in the Main window toolbar.
- **Approach B:** Create menu items specifically for each utility and assign a utility to each menu item. This approach creates a custom utility menu from which the person performing the installation can manually select the utility needed. The utility must run at the time the menu item is selected; therefore, when you develop your map, you must ensure that the menu-item properties for each utility entry in the custom utility menu is set to the **Immediately** setting. If you set the menu-item properties to the **Queue** setting, an error will occur and the

installation process will be terminated. Additionally, a custom utility menu requires a unique structure using links to return to the base menu or to the utility menu after a utility has been run.

There are six steps required to develop this unique structure:

1. Create a menu item specifically for the custom utility menu.

2. Create subordinate menu items for each utility

3. Associate a utility with each menu item in the custom utility menu.

4. Create links under each utility menu item. Each link provides a path after the utility runs that would not otherwise exist. This link usually returns the user to custom utility menu.

5. For each utility menu item, click **View** then **Properties** to open the Menu Item Properties window, click the Submenu tab, and place a check mark in the **Automatically select single item** check box.

6. Create a subordinate menu item (usually something like "Return to the previous menu" or "Exit the utility menu") that enables the person performing the installation to return to the previous menu; then, create a link under it to the previous menu.

The following illustration shows the structure associated with implementing a custom utility menu.
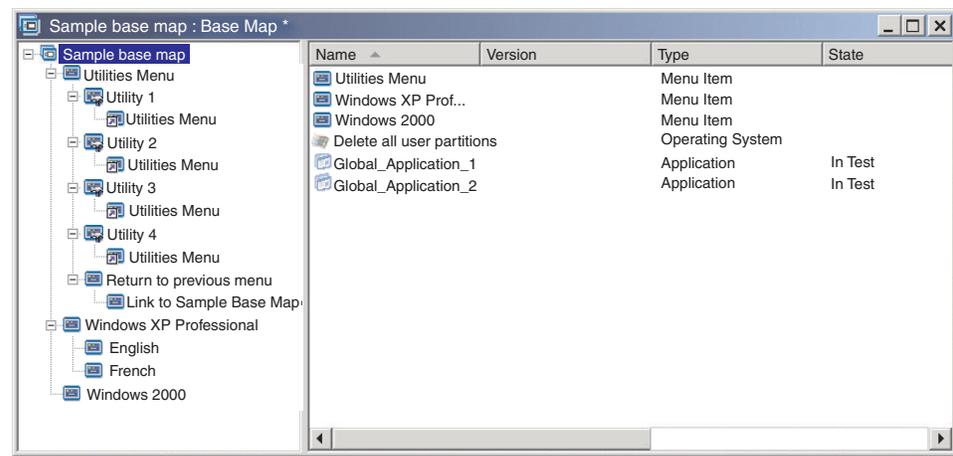
Assigning utilities to base maps

You assign utilities at a map level through the Map Settings window of the base map. The Before/After Menu Options tab enables you to define all of the utilities you want to run before the first installation menu is displayed or immediately after the final selection is made from the final installation menu, but before any modules get installed. Unlike utilities that get assigned to menu items, utilities that get assigned at a map level have no dependencies on specific menu items being selected.

To assign one or more utilities at a map level, do the following:

1. In the left pane of the Base map window, highlight the root entry or any menu item.

2. In the Main window menu bar, click **Tools**, and then click **Settings...**. The Map Settings window opens.

3. Click the **Before/After Menu Options** tab.

4. Click either the **Before user menus** or **After user menus** radio button, depending on when you want the utilities to run.

5. Click the **Insert Utility** icon. The Add Utility window opens.
6. In the Name field, use the drop-down menu to select the utility you want to add.
7. If your repository contains folders, use the left pane of the Add Utility window to navigate to the folder that contains the utility you want to add; then, click **OK**. The Add Utility window closes and the utility is added to the Before/After Menu Options tab.
8. Double-click the utility you just added. The Edit Parameters window opens.
9. In the Parameters field, type any parameters required to run the utility.

   Note: Do not use the default parameters provided in a utility or filter module. Any parameters listed in the module Source tab are listed for example only. In order to update parameters, you need to add the parameters for each utility or filter module in the map settings where the modules are being used.

10. Click **OK**.
11. Repeat steps 4 through 10 for each utility you want to add.
12. When all utilities have been added, Click **OK** in the Map Settings window; then click the **Save** icon in the Main window toolbar.

Using filters in a base map

The use of filters in a base map is optional. The purpose of a filter in a base map is to query the target computer for specific hardware information and then use this information to determine whether a specific menu or menu item should be displayed during the installation process. The hardware information is typically obtained from the computer BIOS. The filters provided with the ImageUltra Builder program enable you to query for the machine type, machine type and model number, platform (desktop or mobile), and whether or not the target computer is an IBM branded computer.

The Edit Parameters dialog is used to specify parameters for filters and utilities. This dialog has been changed to allow the result of a filter to be logically designated as **NOT**:

The **NOT** result checkbox is displayed only when you are editing a filter. By allowing the logical **NOT** operation on the filter's return code, it is no longer necessary to maintain dual sets of filters. For example, suppose you have a filter to check for the existence of a file. If the file exists, the filter returns **1**. If it does not, the filter returns **0**.

Prior to ImageUltra Builder 3.0, you would have had to create a separate filter that would return **1** if the file did not exist, and **0** if it did. With the logical **NOT** check box, you now need only the original filter to check for both situations. A red exclamation mark is displayed next to a parameter listed on the Conditions tab of the Menu Item Properties dialog to indicate that it's result is being logically NOT'd.

When using filters in a map, filters are assigned to menu items, not to modules. The behavior of the filter is controlled through the Menu Item Properties window. Multiple filters can be assigned to a menu item; however, if you assign multiple filters to a menu item, you must define whether only one filter needs to be satisfied or if all filters need to be satisfied for the menu item to be displayed.

The following are a few examples of how you can use a filter in a base map.

Example 1: You have a set of applications that are unique to a mobile computer and you want to use a filter to facilitate the installation of these applications only if the target computer is a mobile computer.

There are several ways to achieve these results depending on the current tree structure in your map. For the purpose of this example, you would create a menu with a single menu item at the end of a branch using the following procedure:

1. Create a menu item at the end of one of your branches and name it "Install mobile applications."
2. Insert all of your mobile-unique applications in the right pane associated with the "Install mobile applications" menu item.
3. Highlight the "Install mobile applications" menu item.
4. In the Main window menu bar, click **View**, and then click **Properties**. The Menu Item Properties window opens.
5. Click the **Conditions** tab and then click the **Insert Filter** icon. The Add Filter window opens.
6. In the left pane of the Add Filter window, click the root entry.
7. In the right pane of the Add Filter window, double-click the **IBM branded Hardware Platform Check** filter. (This filter determines if the target computer is a mobile or desktop computer). The Add Filter window closes and the filter is added to the Conditions tab
8. Double-click the filter you just added. The Edit Parameters window opens.
9. In the Parameters field, type in the "isMobile" parameter, and then click **OK**.
10. Click the **Save** icon in the Main window toolbar.
11. Close the Menu Item Properties window.

When this map is used, the "Install mobile applications" menu item will appear at the target computer only if the target computer is a mobile computer.

Because "Install mobile applications" is the only item in the menu, you can hide the menu and have the mobile-unique applications install automatically by doing the following:

1. Highlight the menu item one level *above* "Install mobile applications"
2. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
3. Click the **Submenu Items** tab.
4. Click the **Automatically select single item** check box so a check mark is visible.

   Note: In the example used throughout this chapter, you would perform steps 1 through 4 for the English and French menu items.
5. In the Main window toolbar, click the **Save** icon.
6. Close the Menu Items Properties window.

The following shows a base map using the Example 1 approach.
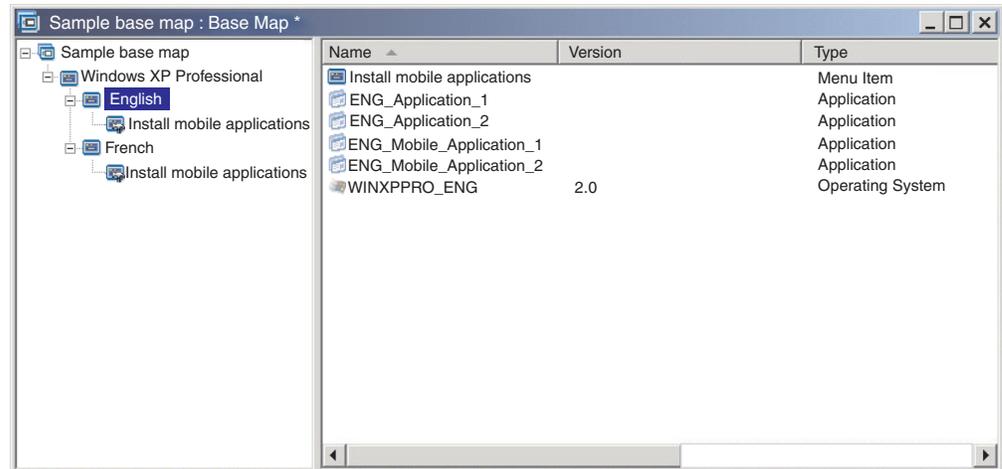
Example 2: You are creating a map that includes a series of operating-system modules that were built from Hardware-Specific Images, and you want to use a filter to display only the menu items that are appropriate for the target computer.

If you did not use a filter, the person performing the installation would have to make a selection from a menu listing all of the machine types that this map supports. By using a filter, the only choices that are presented at the target computer are choices specific to the target computer.

To create a map that includes filters for this example, you would complete the following procedure:

1. Create a menu item below the root entry of the tree for each operating-system module you will include in your map. Give each menu item a name that includes the machine type and operating system.
2. Insert the appropriate operating-system module in the right pane associated with each menu item.
3. Do the following for each menu item:
   a. Highlight the menu item.
   b. In the Main window menu bar, click **View**, and then click **Properties**. The Menu Item Properties window opens.
   c. In the Conditions tab, click the **Insert Filter** icon. The Add Filters window opens.
   d. In the left pane of the Add Filters window, click the root entry.
   e. In the right pane of the Add Filters window, select the **Model Check** filter; then click **OK**. The Add Filters window closes and the filter is added to the Conditions tab.
   f. Double-click the filter you just added. The Edit Parameters window opens.
   g. In the Parameters field, type **checkModel=** followed by the appropriate machine type (for example, **checkModel=2143xxx**, where 2143 is the machine type and xxx is a wildcard for all model numbers), and then click **OK**.
   h. Click the **Save** icon located in the Main window toolbar.
   i. Close the Menu Items Properties window.

When this map is used on a computer having the machine type of 2143, the only selections that will appear at the target computer will be the menu items for the operating system modules developed for that computer. If you had many operating-system modules in your map, but only one Windows XP module and one Windows 2000 module were developed for the target computer, those would be the only two choices shown at the target computer.

If only one operating-system module meets the filter requirements, you can hide the menu and have the correct operating system module install automatically. To enable your map to do this for this example, you would do the following:

1. Highlight the *root entry* of your map.
2. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
3. Click the **Submenu Items** tab.
4. Click the **Automatically select single item** check box so a check mark is visible.
5. In the Main window toolbar, click the **Save** icon.
6. Close the Menu Items Properties window.

The following illustration shows a base map that is constructed to use the Model Check filter. Notice how all of the menu items contain filters. Nested within each menu item is an operating system module made specifically for a computer with the defined model number.
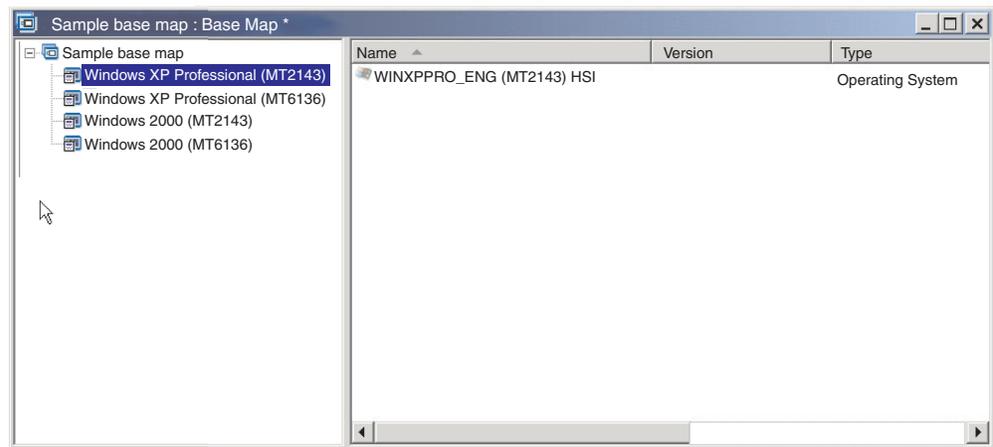
If the target computer is a machine type 2143, the only two menu items that would be displayed at the target computer would be:
• Windows XP Professional (MT2143) HSI
• Windows 2000 (MT2143) HSI

Using links in a base map

A link is a type of shortcut that can be added to base maps only. Links are useful when you have the same menu structure in multiple places within the same map. By using links, you can avoid the redundant task of making the same entries, corrections, or changes in multiple places. For example, assume you have a set of five core applications that get installed on most of the computers you maintain. Within your map, you can define those applications under a menu item once and then link to that menu item from elsewhere in the map when those five core applications need to be defined.

The following illustration shows an example of three menu-item entries that might be used with two different operating systems. Notice how the "Accounting" menu item nested under the Windows 2000 contains a link corresponding "Accounting" menu item nested under the Windows XP Professional entry. In this example, all of the actual application modules are nested under the Accounting, Marketing, and Development entries for the Windows XP Professional entry and links are nested under the corresponding entries for the Windows 2000 entry.
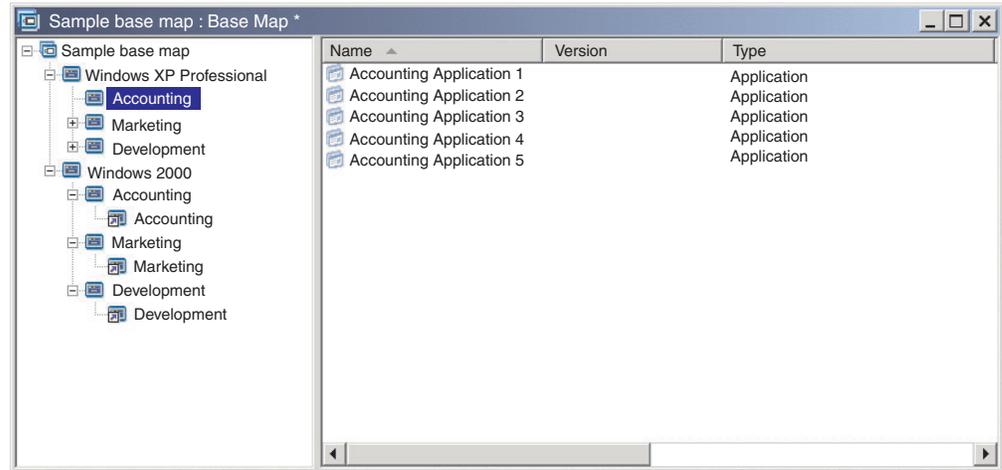
You can also use links to create paths for utilities that get selected from a custom utility menu. For details, see "Using utilities in a base map" on page 86.

Note: Because the menu item descriptions can be repeated in multiple places in a map, such as in the example provided, it is sometimes difficult to determine which menu item is referenced by the link simply by looking at the map. To help determine which menu item is referenced by the link, you can do the following:

1. Highlight the link.
2. From the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
3. Click **menu item** located at the bottom of the window. The referenced link will be highlighted in the right pane of the map window.

Defining user information in a base map

During Windows setup or Sysprep mini-setup, you are typically prompted to provide information for a number of user-unique settings. These settings include:

- Administrator password

  For information on how to change the Administrator password, see "Changing the default administration password" on page 52.

- Computer name
- DHCP configuration (whether or not an IP address is to be obtained from a DHCP configuration)
- DNSServerSearchOrder (IP address of one or more DNS servers. Used when a static IP address will be used instead of obtaining an IP address from a DHCP server)
- User's full name
- Gateway

- IP Address
- Workgroup
- Organization name
- Subnet mask

The ImageUltra Builder program enables you to predefine default settings, or prompt the person performing the installation for specific user information near the beginning of the installation process, or both. Thus, this feature can help minimize the amount of time that an installer needs to be present at the target computer during the installation process.

Note: To minimize the amount of time that an installer needs to be present, you must predefine or establish prompts for *all* user-information settings. The ImageUltra Builder program prompts for user information very early in the installation phase, therefore if all required settings are preset or defined as prompts, all required setup information will be obtained early and the person controlling the installation will not have to be present later in the installation process when Windows setup (or mini setup) normally prompts for this information.

You also can define your own settings and prompts for other purposes. For example, you can prompt the installer for the office number where the computer is being installed and the telephone number for the user. Then, you can run your own software anytime after the installation is complete to collect the office number, telephone number, and user's full name to create or append a company phone book.

User information is implemented in base maps on a map-by-map basis. All predefined user information and installer-provided user information is gathered during the installation process and stored in a file called PERSONAL.INI located in the root folder of the target computer.

To implement user information, do the following:
1. Open the Base Map window.
2. Click the root entry or any menu item in the tree structure.
3. In the Main window menu bar, click **Tools**, and then click **Settings...**. The Map Settings window opens.
4. Click the **User Information** tab.
5. Click the **Add User Information** icon located in the User Information tab, and fill in the appropriate fields. You can use the Information field drop-down menu to work with the typical user settings required by Windows, or type your own setting name in the Information field and define default values and prompts.
6. Click **OK**.
7. Repeat steps 5 and 6 for each setting. When all settings have been defined, continue with the next step.
8. Click **OK**.
9. Click the **Save** icon located in the Main window toolbar.

If you are working with I386 base operating-system modules, it is important that you understand the relationship between the user information settings described in this section and the settings in the UNATTEND.TXT file used with I386 base operating-system modules. Some of the user settings described in this section are a

subset of the settings in the UNATTEND.TXT file and override their counterpart settings in the UNATTEND.TXT file. For more information about the UNATTEND.TXT file, see Chapter 8, "Preparing source files for modules," on page 135.

Controlling the network-sync function

When you create your base map, the New Map Wizard prompts you to define whether or not to enable the network-sync function. You can change the network-sync setting anytime during the map development process. To change the setting, do the following:

1. Open the Base Map window.
2. Click the root entry of the tree structure.
3. In the Main window menu bar, click **Tools**, and then click **Settings...**. The Map Settings window opens.
4. Click the **Network** tab.
5. Use the **Use Network Sync** check box to define the setting.
6. Click **OK**.
7. Click the **Save** icon located in the Main window toolbar.

For more information about the network-sync setting, see "Enabling the network-sync function" on page 24.

Resolve optimal repository

This enables the ImageUltra deploy and net-sync process to select the ideal repository for deployment based on a predefined list of repositories as defined by an IT administrator. The list of repositories is a text-based .INI file that is located in the repository folder. The repository list is keyed by repository description strings, with each key section having corresponding repository information for other repositories.

If a repository list is found, a look-up is performed by searching for the closest repository in the repository list (based on ping response times). If a closer repository is found, the corresponding repository settings will be used for deploying the image. However, if no match is found, or if a repository list is not found, the image is deployed from the current repository.

For an example of a commented Reposlst.ini file refer to the following:

Note: To add a repository to the list, add a new section. Under this section, copy the settings from the PROFILE.INI file from the bootable Network Deploy CD that is created by the ImageUltra Builder Graphical User Interface (GUI).

```
[Settings]
;Uncomment this setting if you want the fastest repository to be selected automatically
;DISPLAY=optimal
;[Description]
;IUSERVER=ServerName
;IUSHARE=ShareName
;IUREPOS=ReposDir
;KEYWORDS=
;CC=US
;KCC=US
;DOMAIN=
;USERID=user
;PASSWD=password
;ENCRYPT=NO
```

```
                      ;UNATTENDED=NO
                      ;BASEMAP=<ID.BMA>
                      ;DRIVERMAP=<ID.DMA>

                      [ImageUltra Repository]
                      IUSERVER=imageultra
                      IUSHARE=data
                      IUREPOS=imageu~1\repos3~1.0
                      DOMAIN=
                      USERID=userid1
                      PASSWD=password
                      ENCRYPT=NO
                      UNATTENDED=No
                      [ImageUltra Repository #2]
                      IUSERVER=imageu-uk
                      IUSHARE=data
                      IUREPOS=imageu~1\repos3~1.0
                      DOMAIN=
                      USERID=userid2
                      PASSWD=password
                      ENCRYPT=NO
                      UNATTENDED=No
```

Controlling the behavior of the service partition

When you create your base map, the New Map Wizard prompts you to define
what action to take on the maps and modules after the image has been installed.
You have three options: Delete none, Delete all, and Delete unused. Your selection
affects the size of the service partition, whether or not console-side recovery will be
available, and the amount of time it will take to complete the installation. See
"Defining service-partition behavior" on page 23 for more information about these
settings.

You can change the setting anytime during the map-development process. To
change the setting, do the following:

1. Open the Base Map window.
2. Highlight the root entry or any menu item in the tree structure.
3. In the Main window menu bar, click **Tools** and then click **Settings...**. The Map
   Settings window opens.
4. Click the **Image Cleanup** tab.
5. Click the **Delete none**, **Delete unused**, or **Delete all** radio button to select the
   action you want to take.
6. Click **OK**.
7. Click the **Save** icon located in the Main window toolbar.

Using a partitioning module in a base map

A partitioning module is a special type of operating-system module that has the
following purposes:

- To delete one or more user partitions before installing an image
- To create an additional data partition on the target computer, beyond the C
  partition
- To preserve one or more pre-existing data partitions on a target computer
- To disable the default partitioning behavior of the ImageUltra Builder program
- To eliminate existing partitions, set up a new single partition, format the new
  partition, and set the new partition as bootable in preparation for Ultra-Portable
  Image installation using an I386 base operating-system module. disk

By default, the ImageUltra Builder program automatically deletes all user partitions on the target computer as a starting point for installation. You can override this default behavior or make the behavior dependent on a menu selection through the use of partitioning modules.

The ImageUltra Builder program provides the following ready-to-use partitioning modules in the ImageUltra folder of the repository:

- **DrivePrep - Delete All Partitions:** This partitioning module is designed to delete all user partitions on the target computer hard disk before the new image is installed.
- **DrivePrep - Delete C Partition Only:** This partitioning module is designed to preserve pre-existing data partitions on a target computer and use the space previously used by the C partition for the new image.
- **DrivePrep - Delete No Partitions:** This partitioning module disables all partitioning activity generated by the ImageUltra Builder program and is used only in special cases with a Portable-Sysprep Image or Hardware-Specific Image that contains all formatting and partitioning instructions.
- **WinXP Home Primary Partition:** This partitioning module for use with Ultra-Portable Images that use I386 base operating-system modules for Windows XP Home only. It deletes existing user partitions, establishes a new single partition using all available space on the hard disk, formats the new partition, and sets the new partition as bootable in preparation for installing the Windows XP Home operating system.
- **WinXP Pro Primary Partition:** This partitioning module for use in Ultra-Portable Images that use I386 base operating-system modules for Windows XP Professional only. It deletes existing user partitions, establishes a new single partition using all available space on the hard disk, formats the new partition, and sets the new partition as bootable in preparation for installing the Windows XP Professional operating system.
- **Win2000 Primary Partition:** This partitioning module is for use with Ultra-Portable Images that use I386 base operating-system modules for Windows 2000 only. It deletes existing user partitions, establishes a new single partition using all available space on the hard disk, formats the new partition, and sets the new partition as bootable in preparation for installing the Windows 2000 operating system.

Note: The three Primary NTFS Partition partitioning modules are for use with I386 base operating-system modules only. Do not attempt to use these modules with any other type of base operating-system module. For details about the Primary NTFS Partition partitioning modules in a map, see "Special considerations for I386 base operating-system modules" on page 84.

You can also create you own partitioning modules to create one or more data partitions and use combinations of partitioning modules in a base map to achieve various results. For detailed instruction on preparing the source files for a partitioning module, see "Preparing source files for a partitioning module" on page 157.

You can insert a partitioning module into a base map in either of the following ways:

- Insert the partitioning module in the right pane of the Base Map window at the root-entry level. You use this implementation when you want the partitioning module to be used on all target computers that use the map, regardless of what menu items are selected from the installation menus.

- Insert the partitioning module in the right pane of the Base Map window at a menu-item level. You use this implementation if you want the use of the partitioning module to be contingent on the selection of a menu item during the installation process. For example, one menu selection can result in deleting all user partitions while a different menu selection can result in deleting the C partition only and preserving any other data partitions that might exist.

The following illustration shows an example of a base map that installs a common data partition to all target computers that use this map.
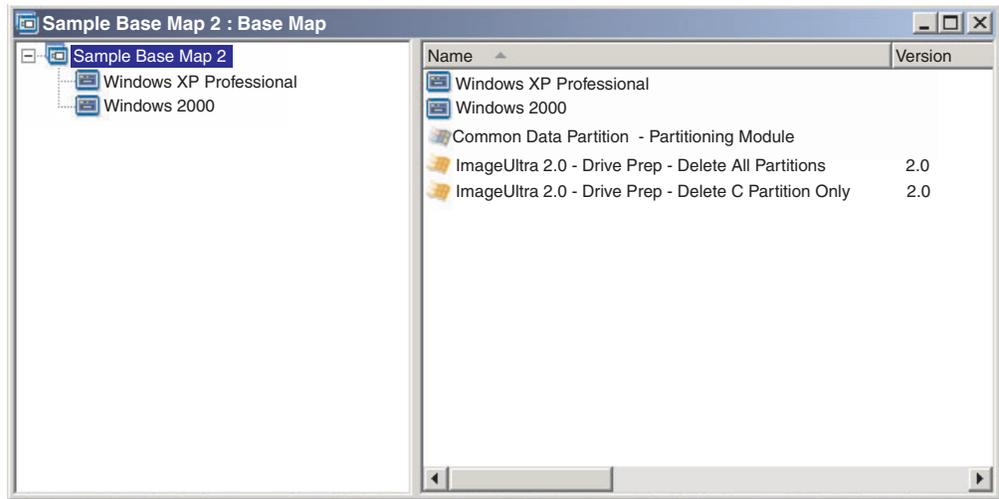
Notice that several partitioning modules are inserted at the root-entry level:
- The ImageUltra-provided partitioning module, *DrivePrep - Delete All Partitions*, deletes all user partitions that are on the target computer.
- The user-developed partitioning module, *Common Data Partition - Partitioning Module*, creates a temporary C partition and installs the data partition. (For information about creating partitioning modules, see "Preparing source files for a partitioning module" on page 157.)
- The ImageUltra-provided partitioning module, *DrivePrep - Delete C Partition Only*, deletes the temporary C partition. All space that was previously used by the temporary C partition is now recognized as free space, which will be used for the remainder of the image.

Note: It is important to understand that the order in which these partitioning modules are installed is extremely important. You set the order in the Install Order tab of the base-map Map Settings window. In the Install Order tab, you select Base Operating System for the module type, move the partitioning modules to the **First** box, and then set the order needed to achieve the desired result. In this example, you would set the order in the **First** box as follows:
1. DrivePrep - Delete All Partitions
2. Common Data Partition - Partitioning Module
3. DrivePrep - Delete C Partition Only

The following illustration shows an example of a base map that installs a data partition only if the "Windows XP Professional" menu item is selected during the installation process.

Note: It is important to understand that the use of any partitioning modules overrides the default ImageUltra partitioning behavior of deleting all partitions. Therefore, in this example, you must insert the *DrivePrep - Delete All Partitions* partitioning module as a starting point.
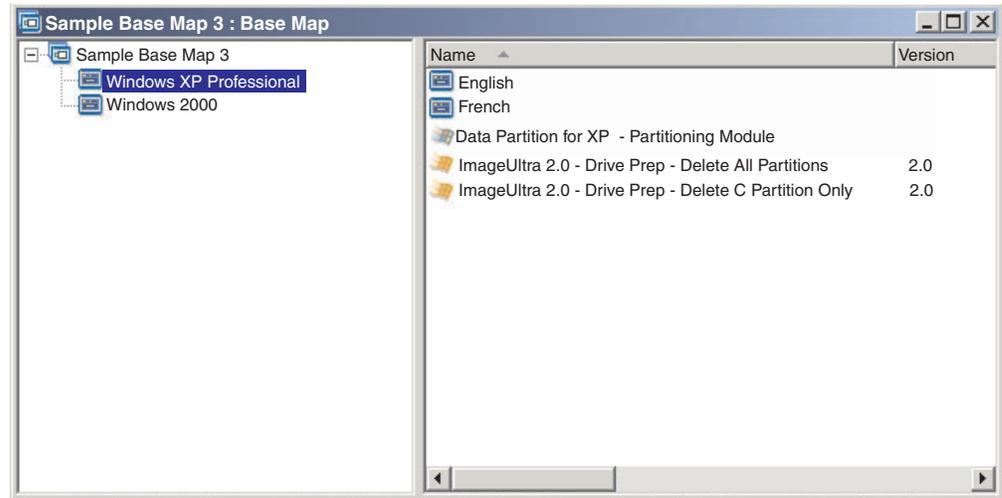
Notice the three partitioning modules that are inserted at the "Windows XP Professional" menu-item level:

- The ImageUltra-provided partitioning module, *DrivePrep - Delete All Partitions*, deletes all user partitions that are on the target computer.
- The user-developed partitioning module, *Data partition for XP - Partitioning Module*, creates a temporary C partition and installs the data partition. (For information about creating partitioning modules, see "Preparing source files for a partitioning module" on page 157.)
- The ImageUltra-provided partitioning module, *DrivePrep - Delete C Partition Only*, deletes the temporary C partition. All space that was previously used by the temporary C partition is now recognized as free space, which will be used for the remainder of the image.

In this example, you world set the order for the partitioning modules in the **First** box of Install Order tab of the Map Settings window as follows:

1. DrivePrep - Delete All Partitions
2. Data partition for XP - Partitioning Module
3. DrivePrep - Delete C Partition Only

The following two illustrations show how the same base map can be used to install different data partitions based on the menu items selected during installation.

```
Sample Base Map 4 : Base Map                                          _ □ ×

□  Sample Base Map 4                    Name  ▲
   └ Windows XP Professional             Data Partition for English - Partitioning Module
        └ English                        ENG_Application_1
        └ French                         ENG_Application_2
   ⊞ Windows 2000                        ImageUltra 2.0 - Drive Prep - Delete All Partitions
                                         ImageUltra 2.0 - Drive Prep - Delete C Partition Only
                                         WINXPPRO_ENG
```

```
Sample Base Map 5 : Base Map                                          _ □ ×

□  Sample Base Map 5                    Name  ▲
   └ Windows XP Professional             Data Partition for French - Partitioning Module
        └ English                        FR_Application_1
        └ French                         FR_Application_2
   ⊞ Windows 2000                        ImageUltra 2.0 - Drive Prep - Delete All Partitions
                                         ImageUltra 2.0 - Drive Prep - Delete C Partition Only
                                         WINXPPRO_FR
```

In this example, you world set the order for the partitioning modules in the **First**
box of Install Order tab of the Map Settings window as follows:

1. DrivePrep - Delete All Partitions
2. Data Partition for French - Partitioning Module
3. Data Partition for English - Partitioning Module
4. DrivePrep - Delete C Partition Only

Note: In this example, the order in which the two user-created modules appear
does not matter as long as the *DrivePrep - Delete All Partitions* module is first
and the *DrivePrep - Delete C Partition Only* is last. Only one of the
user-created modules will be implemented during the install process so they
have no relational dependencies on each other.

The following illustration shows an example of a base map that preserves any
partitions that exist on the target computer beyond the C partition and installs the
new image in the space previously used by the old C partition.
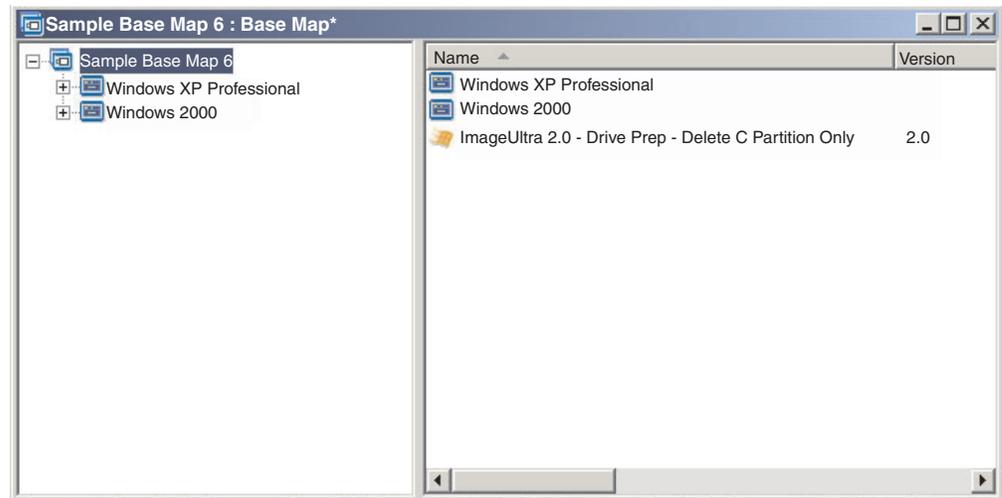
Notice that the ImageUltra-provided module, *DrivePrep - Delete C Partition Only*, is inserted at the root-entry level. In this example, you would ensure that this partitioning module gets installed first by moving the *DrivePrep - Delete C Partition Only* module to the **First** box of the Install Order tab in the Map Settings window. These are the only changes required to preserve pre-existing data partitions on a target computer.

Whenever you use one or more partitioning modules, you must ensure that the partitioning modules are installed in the correct order to achieve the desired result.

Note: When using partitioning modules with an Ultra-Portable Image, the partitioning modules must be installed before any other operating-system modules.

You control the installation sequence of partitioning modules through the Install Order tab of the base-map Map Settings window. To set your partitioning modules to install first, do the following:

1. Open the Base Map window.
2. Highlight the root entry of the tree structure.
3. In the Main window menu bar, click **Tools** and then click **Settings...**. The Map Settings window opens.
4. Click the **Install Sequence** tab.
5. Use the drop-down menu to select **Base Operating System**.
6. Highlight a partitioning module in the **No Preference** area and then click the **Move up** icon to move the partitioning module into the **Install first** area. Repeat this step for each partitioning module.
7. When all partitioning modules have been moved to the **First** area, verify that they are listed in the correct order, and then click **OK**.

Simple and complex base maps

When you develop your base maps, you can develop simple maps that contain a small number of modules, have limited choices, and pertain to one group; or, you can develop complex maps that contain many modules, a larger number of choices, and apply to many groups.

A simple map is constructed to meet the needs of one specific group of users, who have similar requirements. For example, you can construct a simple map designed specifically for an accounting department that consists of one main menu item called "Windows XP with accounting software" and all of the required operating system and application modules inserted under it. The following illustration shows the tree structure associated with a simple map constructed solely for an accounting department.
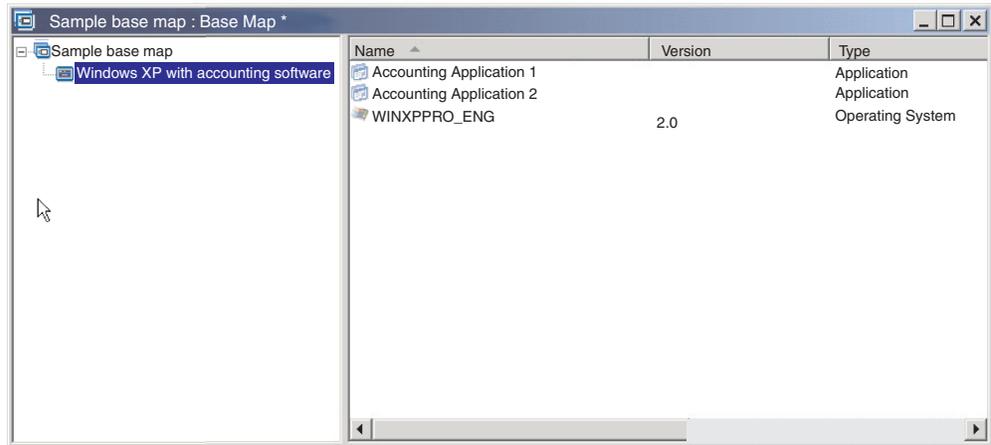
A complex map is constructed to meet the needs of several groups of users, each of which have unique requirements. For example, you can construct a complex map that contains menu items for languages, operating systems, and department names. The following illustration is a conceptual view of a base-map tree structure associated with a complex map constructed for several departments, in several languages. This illustration provides an overall view of how modules can be organized in a complex base map, if you were able to view the map in its entirety.

```
📦 Sample Base Map
  └─ 🖥 Windows XP with legal software
      ├─ 🖥 English (Legal)
      │    ├─ ⊞ WINXPPRO_ENG
      │    └─ 📋 ENG_Application_7
      ├─ 🖥 French (Legal)
      │    ├─ ⊞ WINXPPRO_FR
      │    └─ 📋 FR_Application_5
      └─ 📋 Global_Application_1
  └─ 🖥 Windows XP with sales software
      └─ 🖥 English (Sales)
           ├─ ⊞ WINXPPRO_ENG
           └─ 📋 ENG_Application_2
  └─ 🖥 Windows XP with accounting software
      ├─ 🖥 French (Accounting)
      │    ├─ ⊞ WINXPPRO_FR
      │    ├─ ⊞ WINXPPRO_FR_Service_Pack_1
      │    └─ 📋 FR_Application_1
      └─ 🖥 English (Accounting)
           ├─ ⊞ WINXPPRO_ENG
           ├─ 📋 ENG_Application_1
           └─ 📋 ENG_Application_2
```

Note: This conceptual view of a complex base map does not represent how this map actually looks on the screen. It is provided to help you understand the organization and hierarchy of a complex map.

The following illustration is a view of how this complex base map looks in the interface. The left pane shows the tree structure. The blowouts show the submenu items and modules that would appear in the right pane when a specific menu item is selected.

Name ▲	Version	Type
English (Legal)		Menu Item
French (Legal)		Menu Item
Global_Application_1		Application

Name ▲	Version	Type
ENG_Application_7		Application
WINXPPRO_ENG	2.0	Operating system

Sample Base Map : Base Map *

- Sample Base Map
  - Windows XP with legal software
    - English (legal)
    - French (legal)
  - Windows XP with sales software
    - English (sales)
  - Windows XP with accounting software
    - French (accounting)
    - English (accounting)

Name ▲	Version	Type
FR_Application_5		Application
WINXPPRO_FR	2.0	Operating system

Name ▲	Version	Type
ENG_Application_2		Application
WINXPPRO_ENG	2.0	Operating system

Name ▲	Version	Type
FR_Application_1		Application
WINXPPRO_FR	2.0	Operating system
WINXPPRO_FR_Service_Pack_1		Operating system

Name ▲	Version	Type
ENG_Application_1		Application
ENG_Application_2		Application
WINXPPRO_ENG	2.0	Operating system

Adding a hidden system menu in a base map

A hidden system menu provide a method to make menu items available outside of the normal flow of the installation menus. For example, you might want to make several utilities available through a hidden system menu to help solve diagnostic or recovery issues on the target computer. Instead of including the menu as part of the normal flow, by using a hidden system menu the utilities would be available anytime while the installation menus are active by typing a *system menu keyword*, and the utilities would be accessible only persons authorized to perform the advanced functions associated with the utilities. You are limited to one hidden system menu per base map.

To set a menu as a hidden system menu, open your base map,

1. Open your base map and expand the tree structure.
2. Right-click the menu item that controls the menu that you do not want to appear during the installation process; then, click **Set as System Menu**.
3. Click the **Save** icon located in Main window toolbar.

You assign a *system menu keyword* at a map level. To assign a system menu keyword, do the following:

1. Open your base map.
2. In the Main window menu bar, click **Tools** then click **Settings...**. The Map Settings window opens.
3. Click the **Menu Appearance** tab.
4. In the System Menu Keyword field, type the keyword you want to assign to the hidden system menu. System Menu keywords are case sensitive and limited to 64 alpha-numeric characters; do not use symbols.

5. Click **OK**.
6. Click the **Save** icon in the Main window toolbar.

To access the hidden system menu, type the system menu keyword while an installation menu is active.

Notes:

1. When using the installation menus, do not type a system menu keyword on any menu that is requesting data to be typed. The keyword could be interpreted as data. Instead, move to another menu.
2. Do not press Enter after typing the keyword. Enter might be interpreted as a menu selection.
3. System menu keywords are case sensitive.

Customizing module behavior within a base map

You can alter the behavior of a common module on a map-by-map basis through a new Customization setting. This Customization setting enables you to enter values for an instance of a module in a map that are used by a script that you include as part of your module. During installation, the information in the Customization field is decompressed and stored in the C:\IUWORK*module_name*.DAT file. When the script runs, it picks up the data from the C:\IUWORK*module_name*.DAT file and uses it to perform a specific function.

The information you provide for the Customization setting is free-form text. You can enter a .BAT file, an .INI file, or just a list of values depending on what your script needs. For example, you can:

• Make minor changes to a module without rebuilding it
• Make customized settings for an application, such as predefining a schedule and type of scan for an antivirus program. You are limited only by your understanding of the application and your ability to create a script file for the application.

To use the Customization setting, do the following:

1. Open your map and expand the map tree structure.
2. Locate and highlight the module for which you want to customize the behavior.
3. In the Main window menu bar, click **View** and then click **Properties**. The Properties window opens for the selected module.
4. Click the **Customization** tab.
5. Type the information in the space provided and make a note of the .DAT file name. You will need this file name when you create your script.
6. Click the **Save** icon located in the Main window toolbar.

Special considerations for the Rescue and Recovery product

If you intend to create an image that includes either the Web-downloadable version or retail version of the ThinkVantage Rescue and Recovery product, you must define a custom service partition size that leaves a fixed buffer of at least 500 MB through the Partitions tab of the Base Map settings. For more information, see "Including the ThinkVantage Rescue and Recovery product" on page 179

Previewing your maps

The ImageUltra Builder program has a built-in preview function that simulates the behavior of the installation menus during deployment. You use the preview function at the ImageUltra Builder console. To start the preview function, do the following:

1. Open the Base Map window for the map you want to preview. If the Base Map window is already open, make sure you have saved your latest changes.

2. In the Main window menu bar, click **Tools** and then click **Preview**.

Note: If a map is not constructed correctly, it is possible for the menu system to hang or get in a loop whereby you cannot complete the menu selections in the simulation. To exit from the simulation at anytime, press Ctrl+Esc to return to the Windows desktop; then, close the simulation from the Windows task bar.

Creating a device-driver map tree structure

The New Map Wizard enables you to create a new driver map from scratch or create a new driver map based on an existing driver map. The New Map Wizard handles creating the new driver map entry in the repository, assigning the map a file name, and placing the map in the correct repository location.

There are some fundamental differences between base and driver maps. Unlike base-map tree structures, the only components that are added to driver-map tree structures are menu items and device-driver modules. Menu items in driver maps have no effect on the menus displayed at the target computer. The only purposes for menu items in a driver map are:

- To provide the developer with a mechanism for annotation
- To assign filters to one or more device-driver modules

It is not always necessary to include a driver map as part of a deployable Smart Image. If you deploy an Ultra-Portable Image or a Portable-Sysprep Image to an IBM branded target computer, you have the choice of:

- Using a driver map and managing *all* of the device-driver modules required for that target computer
- Not using a driver map and relying on the ImageUltra-provided device-driver modules in the service partition or HPA of the target computer

If you deploy a Hardware-Specific Image to a target computer, any driver map deployed as part of that Smart Image will be ignored during the installation of the Hardware-Specific Image. The target computer requires all device drivers be included in the Hardware-Specific Image.

If you decide to use driver maps, there are two approaches to constructing them:

- **Create simple driver maps that support a single computer machine type.**

  This approach requires creating and managing multiple maps, each customized with device drivers suited for a particular target computer. If you create a simple driver map, you are responsible for determining *all* of the device drivers that are required for that computer and inserting *all* of the appropriate device-driver modules into the tree structure.

- **Create a complex driver map that supports many computer machine types.**

  This approach uses one menu item for each machine type that your organization supports and then assigns a filter to each menu item to determine which set of

device drivers are to be used during the installation process. This approach requires the use of a filter to determine the machine type. The ImageUltra Builder program provides a filter, *ImageUltra - Model Check*, that you can use for this purpose. (For more information about filters provided with the ImageUltra Builder program, refer to the ImageUltra Builder help system.)

If you create a complex driver map based on an existing map, you can simply make the changes to the tree structure required to support the additional machine type, assign a filter for each new or changed menu item, and define the filter parameters in the Menu Item Properties window for each new or changed menu item. If you create a new complex driver map from scratch, you are responsible for:

– Inserting *all* menu items

– Inserting *all* device-driver modules

– Defining the filter parameters in Menu Item Properties window for *all* menu items

Inserting menu items in a driver map

The use of menu items is required only when you plan to use filters. Unlike base maps, you cannot nest menu items. All menu items in a driver map are inserted at the root level. The following illustration shows a driver-map tree structure with one menu item listed at the root level.
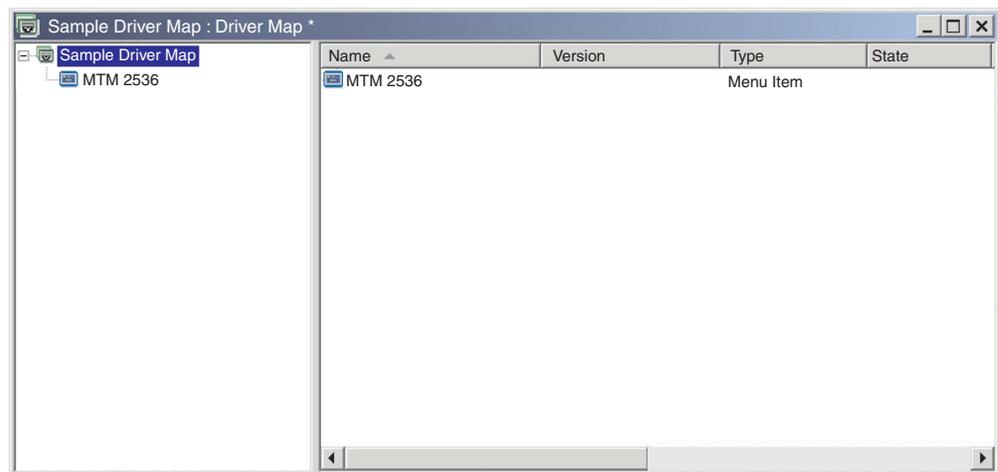
Inserting modules in a driver map

If you are not using menu items in your driver map, device-driver modules are inserted at the root level. If you are using menu items, device-driver modules are inserted in the right pane for the appropriate menu items. A filter impacts only those device-driver modules that are inserted in the right pane of the menu item to which the filter is assigned.

Device-driver modules are the only type of modules that can be added to a driver map. Typically, one or more device-driver modules are inserted in the right pane of each menu item listed in the driver map. The following illustration shows a simple driver-map tree structure with one menu item and a series of associated device-driver modules inserted in the right pane for that menu item.
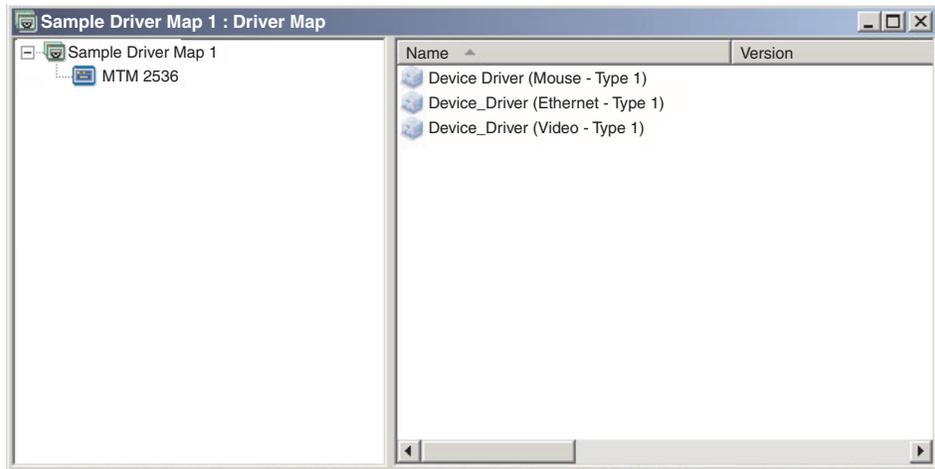
Using filters in a driver map

The purpose of a filter in a driver map is to automatically determine which set of drivers to make available to the Windows setup program (or mini-setup program in the case of a Portable-Sysprep Image) for a specific target computer. The filters provided with the ImageUltra Builder program enable you to query for the machine type, machine type and model number, platform (desktop or mobile), and whether or not the target computer is an IBM branded computer. To reduce the number of menu items and filters in a map, you can provide multiple drivers and let the Windows setup program (or mini-setup program) sort them out based on Plug-n-Play hardware detection.

You can also create your own Win32 filters if you need to query for some other hardware condition. Filters are Win32 programs that run during the installation process. In most cases, filters query the computer BIOS for information, but filters can query any installed hardware that stores identification information in a ROM or EEPROM module.

When using filters in a driver map, filters are assigned to menu items, not to modules. The behavior of the filter is controlled through the Conditions tab of the Menu Properties window. If you assign multiple filters to a menu item, you define whether only one of filters needs to be satisfied or if all of the filters need to be satisfied to use the module.

When a filter is assigned to a menu item, a small condition icon (question mark on a yellow, square background) is displayed in the lower-left corner of the menu-item icon. The following illustration shows the tree structure of a device-driver map that has a filter assigned to each menu item.
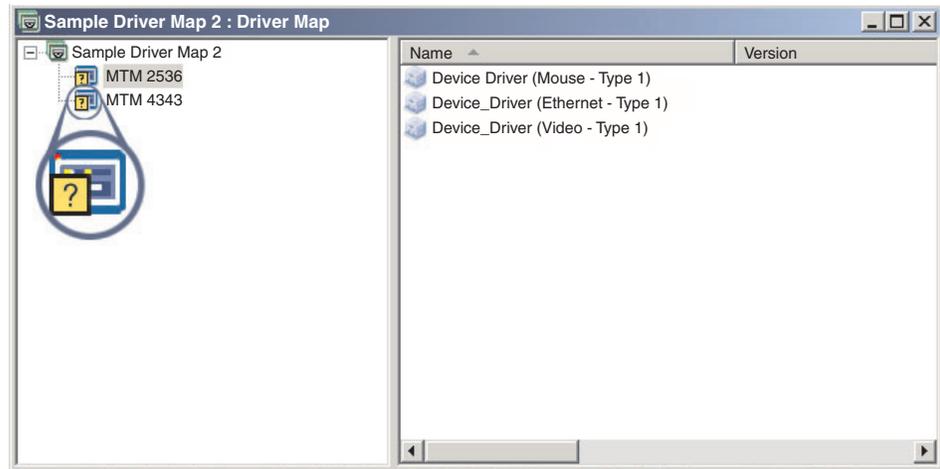
To assign a filter to a menu item in a driver map, do the following

1. Highlight the menu item to which you want to assign a filter.
2. In the Main window menu bar, click **View** and then click **Properties...**. The Menu Item Properties window opens.
3. Click the **Conditions** tab.
4. Click the **Add Filter** icon located in the Conditions tab. The Add Filter window opens.
5. In the Add Filter window, click the filter you want to use and then click **OK**. The Add Filter window closes, a small conditions icon appears at the bottom-left corner of the menu-item icon in the tree structure, and the filter is added to the Conditions tab.
6. Double-click the filter you just added. The Edit Parameters window opens.
7. In the Parameters field, set the parameters you want to use.

   Note: For more information about the format of the parameters associated with the filters provided with the ImageUltra Builder program, see the ImageUltra Builder help system.
   Make sure you remove any example that might be located in the Parameters field.
8. Click **OK**.
9. Close the Menu Items Properties window.
10. Click the **Save** icon located in the Main window toolbar.

Using variables; an advanced base-map function

The use of variables in a base map is optional. Variables are assigned to menu items in a base map; and although variables serve several functions, many variable functions are used in conjunction with each other. For example, it is necessary to assign values to menu items *before* you begin to exercise any of the other variable functions. Once you have assigned variables to menu items you can begin to take advantage of other variable functions. For instance, you can set variables so that the user is prompted for specific information during the installation process. This function enables you to customize the installation process in two ways: by affecting the actually words used for menu choices during the installation process; and by affecting the menu flow during the installation process.

Use variables in association with base map menu items to accomplish any of the following functions:

- Setting a variable to contain a hard coded value
- Setting conditions to determine menu items displayed during the installation process
- Referencing text within menu item descriptions and sub-menu item titles
- Setting a variable to prompt the user for a specific value during the installation process
- Setting a variable so that a prompted value is passed as a parameter for a utility or filter

Note: Variables as described in this section have no association or interaction with the variables used in the User Information tab of the Map Settings window.

Setting a variable to contain a hard coded value

The first step in using variables is to determine whether or not you want to hard code a value to a variable, or if you want to prompt the person performing the installation process for a value to a variable. In some cases, you might want to hard code, or define, a value for a variable used in a base-map menu item. Hard coding values to menu items enables you to develop a structure whereby you can begin to exercise functions that variables provide. For instance, you might have a menu item titled "Windows 2000." In this example, you may want to define the variable for this menu item as "OS." In which case, you could hard code the value for this variable to be "Windows 2000."

To set a variable and its related value for menu items in a base map, do the following:

1. Open the Repository window for the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the map resides.
3. In the right pane of the Repository window, click Base Maps. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. Expand the map tree.
6. Select the menu item to which you want to associate a variable. (In this example, you would select the **Windows 2000** menu item.
7. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
8. In the Variables tab, do the following:
   a. Click the **Insert Variable** icon.
   b. In the Variable field, either type the variable or select a predefined variable using the drop-down menu. (In this example, you would type **OS**).
   c. In the Value field, type the value you want to associate with the variable. (In this example, you would type **Windows 2000**).
   d. Click **OK**. The variable and its associated value are inserted in the Set field on the Variables tab.
   e. Repeat these steps for each variable you want to add.(In this example, you might want to set similar variables for each operating system supported within the base map).
9. Click the Save icon located in the Main window toolbar; then, close the Menu Item Properties window.

The following illustration is a view of two menu items in a base map. The Windows 2000 menu item is a conceptual view, showing that it contains a hard-coded variable and value.
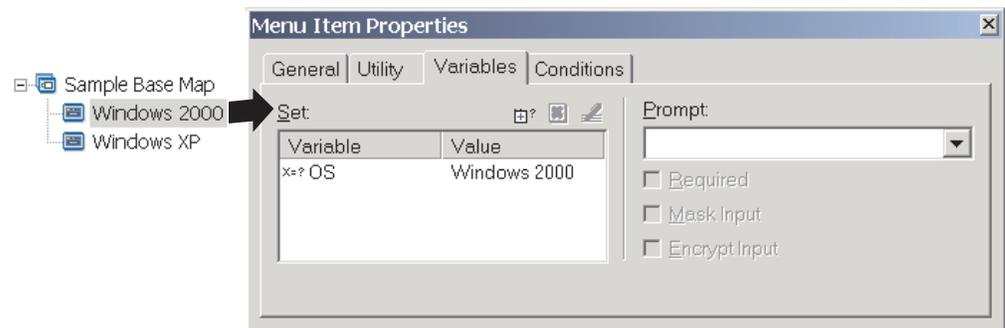
Setting conditions to determine menu items displayed during the installation process

Once key menu items are identified, you can begin to implement other variable functions. For instance, menu item conditions work in conjunction with variables. Conditions enable a base map to evaluate specific values and determine whether or not a specific menu item is displayed during the installation process. For example, you might want to display a specific set of languages if the user chooses "Windows 2000" than you would if the user chooses "Windows XP."

The following example explains how to set a condition so that is evaluates values for a menu item so that different language choices are made available at the target computer during the installation process, depending on the type of operating system that is selected. This example expands on the hard-coded variables example explained in "Setting a variable to contain a hard coded value" on page 110.

Assume that under each operating system menu item, you have a menu item that contains a language list. This menu item, or language list, is titled "Language Menu." Nested within the "Language Menu" menu item are other menu items that contain the titles for all the languages that your base map supports. However, assume that you only want the Spanish menu item to appear if the person performing the installation selects "Windows 2000."

To set a condition so that it evaluates a value for menu items in a base map, do the following:

1. Open the Repository window for the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the map resides.
3. In the right pane of the Repository window, click **Base Maps**. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. Expand the map tree.
6. Select the menu item to which you want to associate a condition. (In the example above, you could select the **Spanish** menu item).
7. In the Main window menu bar, click **View** a nd then click **Properties**. The Menu Item Properties window opens.
8. In the Conditions tab, do the following:

a. Click the **Insert Variable** icon. The Add Variable Condition dialogue box opens.

b. In the Variable field, either type the variable or select a predefined variable using the drop-down menu. (In this example, you would type **OS**).

c. In the Operator field, use the drop-down menu to select either **Equals** or **NOT Equals**. (In this example, you would select **Equals**).

d. In the Value field, type the value you want to associate with the variable. (In this example, you would type **Windows 2000**).

e. Click **OK**.

f. Repeat these steps for each condition you want to add. (In the example above, you might want to set similar conditions for each language used specifically for Windows XP).

9. Click the **Save** icon located in the Main window toolbar; then, close the Menu Item Properties window.

The following illustration is a conceptual view showing how a condition is set for a menu item. In this example, "Spanish" will be displayed as a menu item only if "Windows 2000" is selected.
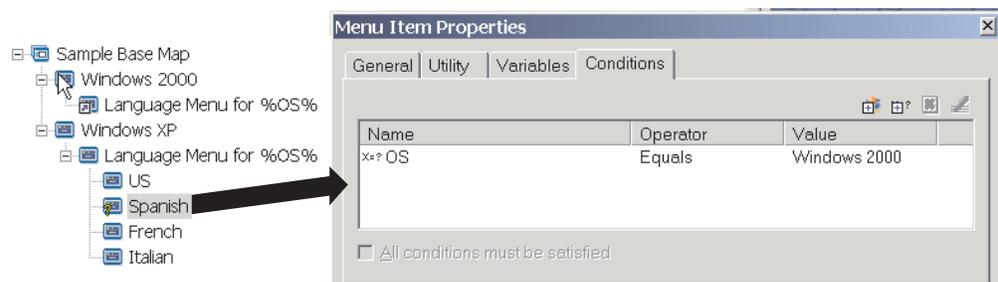
Referencing text within menu items and sub-menu item titles

Variables enable you to write menu item descriptions and sub-menu item titles so that they refer to a specific variable and value. This function enables you to customize menu item descriptions and sub-menu item titles that are displayed during the installation process.

For example, if you were to expand the previous example, you could construct the language list menu item "Language Menu" so that it would display the correct operating system during the installation process, no matter what operating system menu item was selected.

To construct a menu item description so that it dynamically changes the way menu item descriptions are displayed during the installation process, do the following:

1. Open the Repository window for the repository where the map resides.

2. If your repository contains folders, navigate to the folder where the map resides.

3. In the right pane of the Repository window, click **Base Maps**. A list of base maps is displayed.

4. Double-click the appropriate base map. The Base Map window opens.

5. Expand the map tree.

6. Select the menu item to which you want to associate a variable. (In the example above, you would select the **Language Menu** menu item.

7. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.

8. Click the **General** tab.

9. In the Description field, type **Language Menu for %OS%**.

10. Repeat these steps for each description that you want to modify so that it refers to a specific variable and its related value.

11. Click the **Save** icon located in the Main window toolbar; then, close the Menu Item Properties window.

Tip: The same variable can be assigned to different menu items. The values of each variable, however, should be different. This is because values are not set until menu items are selected during the installation process. Once a menu choice is made during the installation process, the variable and its related value are then set and propagated throughout the rest of the map wherever the variable is used.

The following illustration shows how menu items are developed to reference text.
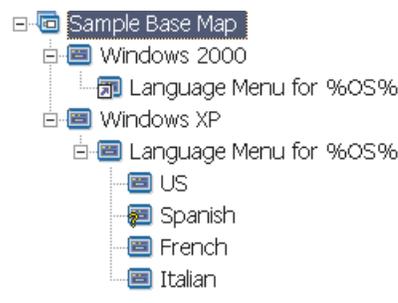
Setting a variable to prompt the user for a specific value during the installation process

Variables can be assigned to menu items so that the person performing the installation is prompted for specific information. That information then becomes the value for a specific variable.

For example, assume that you want to add a variable that prompts for the name of the person performing the installation process. In this case, a menu item could have a variable that prompts for the user's name. The name that the user enters then becomes the value for that variable. So, the value is then propagated throughout the map, wherever the variable is used. This enables the map to reference the user's name throughout the rest of the installation process. This function becomes more significant when maps are designed so that they prompt for information such as: geographical location, division, and department.

There are three different ways that prompted information is processed. First, variables can be set so that prompted answers are required. Second, variables can be set so that prompted answers are masked as they are entered into the computer during the installation process. For instance, if a user is prompted for a User ID, the typed response would appear masked, as in asterisks, when the information is entered. Third, variables can be set so that prompted answers are encrypted once the information is entered.

To set a variable that prompts the user for specific information during the installation process, do the following:

1. Open the Repository window for the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the map resides.
3. In the right pane of the Repository window, click **Base Maps**. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. Expand the map tree.
6. Select the menu item to which you want to associate a variable.
7. In the Main window menu bar, click View and then click Properties. The Menu Item Properties window opens.
8. In the Variable tab, do the following:
   a. Either enter the prompted variable in the Prompt field or use the drop-down menu to select a variable. (In this example, you would type the word **Name** as the variable).
   b. Check the **Required**, **Mask Input**, and/or the **Encrypt Input** check box to determine the way the prompted information is processed.
9. Repeat these steps for each prompted variable that you want to associate with the selected menu item.
10. Click the **Save** icon located in the Main window toolbar; then, close the Menu Item Properties window.

Setting a variable so that a prompted value is passed as a parameter for a utility or filter

A more advanced way to work with variables is to combine variable functions. For instance you can reference a prompted value within a utility or filter parameter. This tasks combines the functions of referencing variables and prompting for values to variables.

Assume that you have an image that requires a certain utility that needs to run uniquely for an image, depending on the department for which the image is intended. In this example, you would set a variable to a menu item that prompts the person performing the installation to enter a department name. The prompted variable could be called" Dept." Then, in the Parameters field for either utilities or filters, you could reference the variable within the parameter string.

To reference a prompted value within a utility or filter parameter, do the following:
1. Open the Repository window for the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the map resides.
3. In the right pane of the Repository window, click Base Maps. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. Expand the map tree.
6. Select the menu item to which you want to reference the variable.
7. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
8. To reference a variable and its associated value within a utility or filter parameter, do the following:

a. Perform the associated steps required to add a utility or filter to a menu item. Except, in the Parameters field, type the variable that you want to reference in the parameter field string. (In this example, you would reference **%Dept.%**).

b. Repeat these steps for each value that you want to reference in the parameter.

9. Click the **Save** icon located in the Main window toolbar; then, close the Menu Item Properties window.

Excluding modules from a map

In base maps, the selection of a menu item menu typically includes all modules associated with that menu item. In driver maps, the association of a filter to a menu item typically defines which modules to include. You can also assign an exclude feature to menu items. For example, you might have a base map that has all of your application modules defined at the root level, and have a *Management* menu item and a *Non-management* menu item. Assume your applications are common for both management and non-management except for two personnel-related tools that are for management use only. In the exclude tab for the Non-management menu item, you can specify that the two personnel-related modules are to be excluded. Then, during the installation process, if the Management menu item is selected, all applications get installed. If the Non-management menu item is selected, all applications get installed except for the two personnel-related applications defined in the Exclude tab.

To exclude menu items, do the following:

1. Open the Repository window for the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the map resides.
3. In the right pane of the Repository window, click either **Base Maps** or **Driver Maps** depending on the type of map you want to modify. Then, double-click the map you want to modify. The Map window opens.
4. In the left pane of the Map window, expand your menu items; then, highlight the menu item you want to associate with the exclude operation.
5. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
6. Click the Exclude tab.
7. In the Exclude tab, click the **Insert Operating System** icon, **Insert Application** icon, or **Insert Device Driver** icon depending on the type of module you want to exclude. The Exclude window opens listing the modules that you can exclude.

   Note: The Exclude window shows all of the modules in your repository, not just the modules currently defined in your map.

8. If your repository contains folders, navigate to the folder that contains the module you want to exclude.
9. Highlight the module you want to exclude and then click **OK**.
10. Repeat steps 7 through 9 for each module you want to exclude. When all excluded modules have been defined, close the Menu Item Properties window and click the **Save** icon located in the Main window toolbar.

Chapter 7. Creating maps

This chapter provides basic information about creating base maps and driver maps as well as how to define the settings associated with each. For additional information, refer to the ImageUltra Builder help system.

Creating and modifying base maps

Base maps control which operating systems, applications, and utilities are deployed as part of a Smart Image. The tree structure of a base map varies depending on the type of image or images contained in the map and the needs of your organization. You must create a map module before you can construct the map tree structure. For detailed information about constructing the map tree structure, see Chapter 6, "Map construction overview," on page 79.

Creating a new base map

This procedure describes how to create a new base map without using an existing base map as a template. To create a new base map, complete the following procedure:

1. Open the repository where you want the new base map to reside.
2. If your repository contains folders, navigate to the folder where you want the new base map to reside.
3. From the Main window menu bar, click **Insert** and then click **Base Map...**. The New Map Wizard opens.
4. Click **Next**.
5. Verify that the **Base Map** radio button is selected, and then click **Next**.
6. In the Name field, type the name you want to use to identify your new base map. Be as specific as possible. This name will be used to identify your base map in the repository.
7. Click **Next**.
8. Click **Next** without selecting any maps.
9. If you want the target computer to do a network-sync operation (check for updated maps before installation), click the **Yes** radio button; if not, click the **No** radio button.
10. Click **Next**.
11. Do one of the following:
    - If after installation you want the service partition of the target computer to contain the complete Smart Image, click the **Delete none** radio button.
    - If after installation you want the service partition of the target computer to contain only the maps and modules required to recreate the installed image, click the **Delete unused** radio button.
    - If after installation you do not want any maps or modules contained in the service partition of the target computer, click the **Delete all** radio button.
12. Click **Next**.
13. Click **Finish**. The New Map Wizard closes and the Base Map window opens.

For more information about working with the tree structure, see Chapter 6, "Map construction overview," on page 79.

For step-by-step instructions on working with other aspects of a base map, see the following:

- "Adding menu items to a base map"
- "Adding operating-system modules to a base map" on page 119
- "Adding application modules to a base map" on page 120
- "Working with base-map settings and menu-item properties" on page 121

Creating a new base map based on an existing base map

In most cases, when you finish testing a map, you promote it to the gold state. Once the map is in the gold state, it is locked and you can no longer make modifications to the map tree structure, map settings, or the menu-item properties. However, you can make a copy of the map, make changes to the copy, test and promote the copy, and then update the network-sync settings to use the modified copy in place of the original map.

Or, you might want to create a new map that has contents similar to an existing map. In this case, you can make a copy of an existing map, make changes to the copy, and then begin your testing process.

This procedure describes how to create a new base map using an existing base map as a template. The new base map will contain all of the module entries and settings that were contained in the source base map. To create a new base map based on an existing base map, complete the following procedure:

1. Open the repository where you want the new base map to reside.
2. If your repository contains folders, navigate to the folder where you want the new base map to reside.
3. From the Main window menu bar, click **Insert** and then click **Base Map...**. The New Map Wizard opens.
4. Click **Next**.
5. Verify that the **Base Map** radio button is selected, and then click **Next**.
6. In the Name field, type the name you want to use to identify your new base map. Be as specific as possible. This name will be used to identify your base map in the repository.
7. Click **Next**.
8. Select the base map you want to use as the source and then click **Next**.
9. Click **Finish**. The Base Map window opens for the new base map.

Adding menu items to a base map

To add menu items to a base map, complete the following procedure:

1. Open the repository where the base map resides.
2. If your repository contains folders, navigate to the folder where the base map resides.
3. In the right pane of the Repository window, click the **Base Maps** tab. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.

   Note: When inserting a menu item, the new menu item will be one level subordinate to the entry (root entry or other menu item) that you select in the next step.

5. Do one of the following:

- If the map tree already has menu items in it, expand the map tree and click the entry one level above the location where you want to add the new menu item.
- If the map tree does not have any menu items in it, click the root entry.

6. In the Main window menu bar, click **Insert** and then click **Menu Item**. A "New Item" box is added to the tree.

7. In the "New Item" box, type the text for the new menu item and then press Enter.

8. Repeat steps 5 through 7 until all menu items are added.

9. To move a menu item up or down, click on the menu item; then, click the **Move Item Up** or **Move Item Down** icon located in the Main window toolbar.

10. When you are finished, click the **Save** icon in the Main window toolbar.

Note: To delete modules or menu items within a map, select the module or menu item and click the **Delete** icon located in the Main window toolbar.

Adding operating-system modules to a base map

To add an operating-system module to a base map, complete the following procedure:

1. Open the repository where the map resides.

2. If your repository contains folders, navigate to the folder where the base map resides.

3. In the right pane of the Repository window, click the **Base Maps** tab. A list of base maps is displayed.

4. Double-click the appropriate base map. The Base Map window opens.

5. Expand the map tree.

6. Click the menu item to which you want to add the operating-system module.

7. In the Main window menu bar, click **Insert** and then click **Operating System...**. The Add Operating Systems window opens.

8. In the left pane of the Add Operating Systems window, navigate to the folder that contains the operating-system module you want to insert.

9. In the right pane of the Add Operating Systems window, click the module you want to add. If you want to add more than one module at a time, use the Shift key or Ctrl key in conjunction with the mouse to highlight multiple modules.

   Note: If you imported modules from an IBM branded computer, you might have one or more operating-system modules in your repository with names beginning with "ImageUltra Customization for..." or "ImageUltra Customization Module... ". Never add one of these modules or the ImageUltra 2.2 - ImageUltra Customizations module in a branch where it will be installed as part of a Portable-Sysprep image.

   If you are creating a base map for an Ultra-Portable Image and are adding an I386 base operating-system module, then you must include the ImageUltra Customization module, the Sysprep module, the appropriate Primary DOS Partition partitioning module to accommodate the operating system your are installing, and an UNATTEND.TXT module that you created to the base map.

10. Click **OK**.

11. Repeat steps 6 through 9 for each operating-system module you want to add.

12. When you are finished, click the **Save** icon in the Main Window toolbar.

Adding application modules to a base map

To add an application module to a base map, complete the following procedure:

1. Open the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the base map resides.
3. In the right pane of the Repository window, click the **Base Maps** tab. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. Expand the map tree.
6. Click the menu item to which you want to add the application module.
7. In the Main window menu bar, click **Insert** and then click **Application...**. The Add Applications window opens.
8. In the left pane of the Add Applications window, navigate to the folder that contains the application module you want to insert.
9. In the right pane of the Add Applications window, click the module you want to add. If you want to add more than one module at a time, use the Shift key or Ctrl key in conjunction with the mouse to highlight multiple modules.
10. Click **OK**.
11. Repeat steps 6 through 10 for each application module you want to add.
12. When you are finished, click the **Save** icon in the Main Window toolbar.

Customizing module behavior within a base map

You can alter the behavior of a common module on a map-by-map basis through a new Customization setting. This Customization setting enables you to enter values for an instance of a module in a map that are used by a script that you include as part of your module. During installation, the information in the Customization field is decompressed and stored in the C:\IUWORK\module_name.DAT file. When the script runs, it picks up the data from the C:\IUWORK\module_name.DAT file and uses it to perform a specific function.

The information you provide for the Customization setting is free-form text. You can enter a.BAT file, an .INI file, or just a list of values depending on what your script needs. For example, you can:

- Make minor changes to a module without rebuilding it
- Make customized settings for an application, such as predefining a schedule and type of scan for an antivirus program. You are limited only by your understanding of the application and your ability to create a script file for the application.

To use the Customization setting, do the following:

1. Open your map and expand the map tree structure.
2. Locate and highlight the module for which you want to customize the behavior.
3. In the Main window menu bar, click View and then click Properties. The Properties window opens for the selected module.
4. Click the Customization tab.
5. Type the information in the space provided and make a note of the .DAT file name. You will need this file name when you create your script.

6. Click the Save icon located in the Main window toolbar.

Working with base-map settings and menu-item properties

During the installation process on a target computer, the person performing the installation typically makes selections from a set of menus to specify the image that is to be installed. The overall appearance of these menus and the overall behavior of the map are controlled through the base-map settings. The individual menu items and the filters and modules associated with each menu item are controlled through the base-map properties. Detailed information about working with the base-map settings and menu-item properties can be found in Chapter 6, "Map construction overview," on page 79.

The following procedures describe how to specify new settings and properties or modify existing settings and properties. To use these procedures, you must have already created your base map and set up your map tree structure.

Specifying base-map settings

Base-map settings control the overall appearance of the menu system and the overall behavior of the installation. To specify or change a base-map setting, do the following:

1. Open the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the base map resides.
3. In the right pane of the Repository window, click the **Base Maps** tab. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens.
5. In the Main window menu bar, click **Tools** and then click **Settings...**. The Map Settings window opens.
   - In the General tab, you can do the following:
     – If you want to add comments to the map, you can do so by typing free-form text in the Comments field. The Word Wrap check box enables you to automatically wrap text within the confines of the Comments field or establish your own line length and use a horizontal scroll bar.
     – If you want to add keywords to your map, you can do so by typing one or more keywords in the Keywords to Associate with this Map field. You can use any characters in a keyword except for the vertical bar (|) and the ampersand (&). Separate keywords with a space.

       Note: Keywords enable the developer to create a *Deployment Disc* (CD or DVD) that limits the selection of maps to only those that have one or more matching keywords. The ImageUltra Builder program automatically assigns some keywords to maps based on the certain map settings and the build status. You can manually assign other keywords to maps through the Map Settings window. For more information about predefined keywords and how keywords are used, refer to the ImageUltra Builder help system.
   - To predefine user information with default values or prompt the installer for user information near the beginning of the installation process, do the following in the User Information tab:

     Note: If you are deploying an Ultra-Portable Image based on an I386 base operating-system module, some of the user settings defined in the

following steps will override settings defined in the UNATTEND.TXT file associated with the I386 base operating-system module.

a. Click the **Insert User Information** icon located in the User Information tab.

b. In the Information field, use the drop-down menu to select the type of setting you want to predefine or for which you want to prompt.

c. If you want to predefine a value for this setting, type the information in the Default field.

d. If you want to prompt the person performing the installation for a value for this setting, click the **Display the following prompt** check box, so a check mark is visible; then, in the space provided, type the prompt you want displayed.

e. Click **OK**.

f. Repeat these steps for each type of setting.

- To automatically run one or more utilities *before* the menu system is displayed at the target computer or *after* the final selection is made from the final menu, do the following in the Before/After Menu Options tab:

a. Click either the **Before user menus** or **After user menus** radio button, depending on when you want the utilities to run.

b. Click the **Insert Utility** icon in the Before/After Menu Options tab. The Add Utility window opens.

c. In the Add Utility window, navigate to the utility you want to use, click the utility, and then click **OK**. The Add Utility window closes and the utility is added to the Before/After Menu Options tab.

d. Double-click the utility you just added. The Edit Parameters window opens.

e. In the Parameters field, type any parameters required by the utility.

f. Click **OK**.

g. Repeat these steps for each utility you want to add.

- To control the order in which either customer-first-boot modules or base operating-system modules are installed, do the following in the Install Sequence tab:

a. Use the drop-down menu to select the type of modules you want to sequence.

   Note: Customer-first-boot modules and Base operating-system modules are the only modules for which you can set the sequence at a map level. Customer-first-boot modules appear as selections only if the install hook attribute of the module was defined as **Install during customer first boot** in the Options tab module window.

b. Highlight the module you want to move.

c. Use the **Move Item Up** or **Move Item Down** icon to move the module up or down in the sequence.

   Note: Move modules that need to be installed first into the **First** box, then set the sequence within the **First** box. Move modules that need to be installed last in the **Last** box, then set the sequence within the **Last** box. In most cases, partitioning modules need to be installed before other operating-system modules.

d. When you are finished setting the sequence, click **OK**.

- To define which files get stored in the service partition of the target computer *after* the image is installed, select one of the following from the Image Cleanup tab.
  - If after installation you want the service partition of the target computer to contain the complete Smart Image, click the **Delete none** radio button.
  - If after installation you want the service partition of the target computer to contain only the maps and modules required to recreate the installed image, click the **Delete unused** radio button.
  - If after installation you do not want any maps or modules contained in the service partition of the target computer, click the **Delete all** radio button.
- To set the network-sync function do the following in the Network tab.
  - Place a check mark in the **Use Network Sync** check box to enable the network-sync function. The network-sync function checks for updated maps during the deployment and installation processes.
  - Remove the check mark from the **Use Network Sync** check box to disable the network-sync function.
- To select the method of deployment and installation, do the following the Network tab:
  - Click the **Yes** radio button to configure the map for a direct-network installation. A direct-network installation deploys and installs the image directly over the network. The target computer requires a network connection throughout the entire process.
  - Click the **No** radio button to configure the map for a standard deployment with local installation. This option deploys the Smart Image to the service partition of the target computer either over the network or from a *ImageUltra Builder Distribution Disc* (CD or DVD) and the installation takes place locally without any dependencies on a network connection.
  - Click the **Prompt the user when the image is deployed** radio button to prompt the user to select the type of installation as soon as the map has been selected.
- To control window settings of the installation menus, do the following in the Menu Window tab:
  - In the Window title field, type the title or description of the map that will appear in the title bar area of the map viewer window.
  - In the Window size area, select how the menu information is presented during installation. Click either Minimized, Maximized, or Dimensions radio button. Use this option to customize the dimensions and size of the map viewer window. The Width and Height drop-down boxes enable you to specify the dimension in units of pixels or as a percentage of the screen size.
  - In the Window control area, check the Hide title bar buttons, Disable ALT +F4 key, show scrollbars, Disable ESC key, or the Resizeable check box to customize the map viewer window controls.
  - In the Align buttons area, use the drop-down box to customize how the Back, Next, and Cancel buttons will appear along the bottom of the map viewer window. Select Right, Left, or Center.
- To control the appearance of the installation menus, do the following in the Menu Appearance tab:
  - In the Color Settings area, use the **Item** drop-down menu to select an item, then; use the **Color** drop-down menu to select an color for that item.

- In the Layout Settings area, set your background character, margins, 3D shadowing, select-line text, and status-line text.
- In the Keyboard Settings area, use the drop-down menus to select which key to use to navigate to a previous menu and which key to use to exit from the menu system. You also can enable or disable the Ctrl+Alt+Del key sequence and set a keyword to get to a hidden system menu.
- To set a custom size for the service partition, do one of the following in the Partition tab:
  - To set a fixed size for the partition, click the **A Fixed size** radio button; then, type the size of the partition (in megabytes) in the space provided.
  - To set a partition size based on a percentage of the drive, click the **A percentage of the drive** radio button; then, use the slider bar to select a percentage.
  - To set a partition size with a fixed-size buffer, click the **Based on size of image** radio button. Then, put a check mark in the **Add an additional cushion** check box and type the buffer size in the space provided. To set a maximum partition size, place a check mark in the **Partition size may not exceed** check box and use the slider bar to select a percentage of the drive.
6. When you have completed making all of your changes in the Map Settings window, click **OK**; then, click the **Save** icon in the Main window toolbar.

Adding a hidden system menu in a base map

A hidden system menu provides a method to make menu items available outside of the normal flow of the installation menus. For example, you might want to make several utilities available through a hidden system menu to help solve diagnostic or recovery issues on the target computer. Instead of including the menu as part of the normal flow, by using a hidden system menu, the utilities would be available anytime while the installation menus are active by typing a system menu keyword, and the utilities would be accessible only by persons authorized to perform the advanced functions associated with the utilities. You are limited to one hidden system menu per base map.

To set a menu as a hidden system menu, and do the following:
1. Open your base map and expand the tree structure.
2. Right-click the menu item that controls the menu that you do not want to appear during the installation process; then, click Set as System Menu.
3. Click the Save icon located in Main window toolbar.

You can assign a system menu keyword at a map level. To assign a system menu password do the following:
1. Open your base map.
2. In the Main window menu bar, click Tools then click Settings.... The Map Settings window opens.
3. Click the Menu Appearance tab.
4. In the System Menu Keyword field, type the keyword you want to assign to the hidden system menu. System Menu keywords are case sensitive and limited to 64 alpha-numeric characters; do not use symbols.
5. Click OK.
6. Click the Save icon in the Main window toolbar.

Specifying menu-item properties in a base map

Each menu item in a map, including the root entry, has a unique set of menu-item properties associated with it. The menu-item properties control the following:

- **General:** This property controls the following:
  - The text of the menu item or root entry. If you change the text of the root entry, you also change the name of the map in the repository
  - The alignment of the menu items when viewed in the installation menus. You can independently define each menu item to be left-justified, right-justified, or centered.
  - Whether or not the menu item should remain in a map if the **Delete unused** option is selected in the Image Cleanup tab of the Map Settings window and the image can be installed without selecting that specific menu item. For example, if you have a menu item for a utility, but during the installation process the selection of that menu item is optional, you can place a check mark in the **Show always** check box to ensure that the menu item will always be available if the image is reinstalled from the service partition. If the **Show always** check box is not used, and the image is reinstalled from the service partition, the only menu items that will be available are the menu items that were selected during the original installation.

- **Submenu Items:** This property is available only if there are submenu items nested under the selected menu item. The Submenu-items property controls the following:
  - The text in the title line of the menu
  - The spacing between menu items
  - Whether a menu should be displayed if there is only one menu item listed or if the single menu item should be automatically selected
  - Whether or not multiple menu items can be selected

- **Utility:** This property is available for menu items only; it is not available for the root entry. You use the Utility property to assign a utility to a menu item.

- **Variables:** You use the Utility property to assign a variable to a menu item or to the root entry.

- **Conditions:** This property is available for menu items only; it is not available for the root entry. You use the Conditions property to assign one or more filters or variable conditions to a menu item.

- **Exclude:** This property enables you to select modules from your base map that are to be excluded from the installation process if the associated menu item is selected. For example, you might have all of your application modules defined at the root level of your base map, and have a *Management* menu item and a *Non-management* menu item. Assume your applications are common for both management and non-management except for two personnel-related tools that are for management use only. In the exclude tab for the Non-management menu item, you can specify that the two personnel-related modules are to be excluded. Then, during the installation process, if the *Management* menu item is selected, all applications get installed. If the *Non-management* menu item is selected, all applications get installed except for the two personnel-related applications defined in the Exclude tab.

To define base-map properties, do the following:

1. Open the repository where the map resides.
2. If your repository contains folders, navigate to the folder where the base map resides.

3. In the right pane of the Repository window, click the **Base Maps** tab. A list of base maps is displayed.
4. Double-click the appropriate base map. The Base Map window opens with the root entry highlighted.
5. Expand the map tree.
6. Select the root entry or a menu item.
7. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
8. Click the **General** tab and do the following:
   a. In the Description field, verify that the text is correct for the selected menu item (or root entry).
   b. In the Justify field, use the drop-down menu to select the alignment for the selected menu item.
   c. In the **Show always** check box, add a check mark only if you want the menu item to be included in menu system if the image is reinstalled from the service partition. This check box has meaning only if the **Delete unused** option is selected in the Image Cleanup tab of the Map Settings window.
9. Click the **Submenu Items** tab and do the following:

   Note: The Submenu items tab is present only if submenu items are nested under the selected menu item.

   a. In the Title field, type a title or instruction that applies to the submenu items. For example, if your menu contains a list of operating systems, you might want the title to be "Select an operating system."
   b. In the Spacing field, set the number of blank lines that you want to use to separate menu items.
   c. Place a check mark in the **Automatically select single item** check box if you want the installation program to automatically make the selection if only a single menu item is listed. A single-menu-item condition will exist if only one submenu item is listed under the selected menu item in the tree structure, or if you assigned a filter to multiple submenu items and only one submenu item satisfies the condition of the filter.
   d. Place a check mark in the **Allow multiple selections** check box if you want to enable the person controlling the installation to choose more than one item from the menu. For example, you might want to provide the ability to choose one or more items from a list of application programs.
10. Click the **Utility** tab if you want to associate a utility with the menu item, then do the following:
    a. In the Name field, use the drop-down menu to select a utility from the module repository.
    b. In the Parameters field, type any parameters required to run the utility.
    c. In the Run area, click one of the following radio buttons:
       - **Immediately:** The utility runs as soon as the menu item is selected at the target computer.
       - **Queue:** When the menu item is selected the utility is placed in a queue until the final selection is made from the final menu. Then, all utilities in the queue are run before any modules are installed.
11. Click the **Variables** tab if you want to define a variable; then, do the following:

Note: For additional details about using variables in a map, see "Using variables; an advanced base-map function" on page 109.

a. Click the **Insert Variable** icon. The Set Variable window opens.

b. In the Variable field, use the drop-down menu to select a predefined variable or type a new variable.

c. In the Value field, type the value you want to associate with the variable.

d. Click **OK**. The new variable is added to the list of variables.

e. If you want to prompt the person performing the installation for input, type the prompt text in the **Prompt** field. Then, do any of the following:

- Place a check mark in the **Required** check box if this input is required.
- Place a check mark in the **Mask Input** check box if you want the input to be hidden as it is typed.
- Place a check mark in the **Encrypt Input** check box if you want the input to be encrypted.

12. Click the **Conditions** tab if you want to associate a filter or variable condition with the menu item.

To associate a filter with a menu item, do the following:

a. In the Conditions tab, click the **Insert Filter** icon. The Add Filters window opens.

b. If the filter you want to use is in a folder, navigate to that folder.

c. Select the filter from the repository; then, click **OK**. The filter is added to the list.

d. Double-click the filter you just added. The Edit Parameters window opens.

e. In the **Parameters** field, type any required parameters. For information about the parameters for the filters provided with the ImageUltra Builder program, see the ImageUltra Builder help system.

f. Click **OK**.

g. Repeat these steps for each filter you want to add.

To associate a variable condition with a menu item, do the following:

a. In the Conditions tab, click the **Insert Variable** icon. The Add Variable Condition window opens.

b. Use the Variable drop-down menu to select the variable for which you want to set the condition.

c. Use the Operator drop-down menu to select either **Equals** or **NOT Equals**, depending on the condition you want to set.

d. In the **Value** field, type the value you want to associate with the variable.

e. Click **OK**. The variable is added to the list.

If you have multiple filters or variables listed, or any combination of filters and variables, do one of the following:

- If you want the menu item to be displayed only if the conditions set by all filters are met, place a check mark in the **All conditions must be satisfied** check box.
- If you want the menu item displayed if the conditions set by at least one filter is met, *do not* place a check mark in the **All conditions must be satisfied** check box.

13. Click the **Exclude** tab if you want specific modules to be excluded when this menu item is selected, then do the following:

a. In the Exclude tab, click either the **Insert Operating System** icon, **Insert Application** icon, or **Insert Device Driver** icon depending on the type of module you want to exclude. The appropriate Exclude window opens.

b. If the module you want to exclude in a folder, navigate to that folder.

c. Click the module you want to exclude, then click OK. The excluded module is added to the list in the Exclude tab.

d. Repeat these steps for each module you want to exclude.

14. When you are done modifying the attributes, click the **Save** icon in the Main window toolbar; then, close the Menu Item Properties window.

Dynamically excluding modules from a map

One benefit of creating a custom base or driver map is to control which modules are deployed as part of a Smart Image. Typically, both base maps and driver maps are designed to accommodate a variety of users within your organization. It is possible to create a more "streamlined" base or driver map by setting the properties of a menu item so that certain modules are excluded from the installation process. The ability to exclude specific modules from a menu item during the installation process enables you to create more flexible maps that are deigned to accommodate a variety of users.

To exclude modules from a menu item that is used in either a base map or driver map do the following:

1. Open the Repository window for the repository where the map resides.

2. If your repository contains folders, navigate to the folder where the map resides.

3. In the right pane of the Repository window, click either base map or driver map, depending on the map that contains the menu item and modules you want to exclude. A list of maps is displayed.

4. Double-click the appropriate map. The map window opens for the selected map.

5. Highlight the menu item that contains modules you want to exclude during the installation process.

6. Using the toolbar, Click View; then click Properties. The Menu Item Properties window opens.

7. Click the Exclude tab.

8. Click the Operating System, Application, or Device Driver icon, depending on the type of module you want to exclude from the menu item. The Exclude window opens, displaying a list of the appropriate modules that exist in the repository.

   Note: Even though all modules of a specific type appear in the Exclude window, you only want to select modules that are currently contained in the menu item. For example, if you want to exclude a specific application from a base map menu item, you are given a choice of all application modules that exist in your repository when the Exclude window opens. However, you only want to select the desired application module that is specified in the menu item. Typically, not all application modules listed in the repository are also listed in the menu item.

9. Click the module, or modules, that you want to exclude from the menu item; then, click OK.

Creating and modifying driver maps

Driver maps control which device drivers and hardware-specific applications are deployed as part of a Smart Image. Driver maps can be simple (specific to one machine type) or complex (designed for many machine types), depending on the needs of your organization.

There are three approaches to creating a driver map:

- If you import device-driver modules from a source computer, the ImageUltra Builder program automatically creates a simple driver map for you based on the modules you import. You can use this device-driver map as is, modify the device-driver map to suit your needs, or make a copy of the map and modify the copy.
- You can use the New Map Wizard to create a driver map from scratch.
- You can use the New Map Wizard to create a driver map based on an existing driver map, and then make whatever modifications are needed.

You can also replace an existing map with a modified copy by updating the Network Sync Table.

Detailed information about constructing the map contents is in Chapter 6, "Map construction overview," on page 79. The information in this section describes the methods used to create and modify maps.

Creating a new driver map

Use the following procedure to create a new driver map from scratch:

1. Open the repository where you want the new driver map to reside.
2. If your repository contains folders, navigate to the folder where you want the new driver map to reside.
3. From the Main window menu bar, click **Insert** and then click **Driver Map...**. The New Map Wizard opens.
4. Click **Next**.
5. Verify that the **Driver Map** radio button is selected, and then click **Next**.
6. In the Name field, type the name you want to use to identify your new driver map. Be as specific as possible. This name will be used to identify your map in the repository.
7. Click **Next**.
8. Click **Next** without selecting any maps.
9. If you want to enable the network-sync function, click the **Yes** radio button; if not, click the **No** radio button.
10. Click **Next**.
11. Click **Finish**.
12. The New Map Wizard closes and the Driver Map window opens.

For information about working with the tree structure, see Chapter 6, "Map construction overview," on page 79.

For step-by-step instructions on working with other aspects of a driver map, see the following:

- "Adding menu items to a driver map" on page 130
- "Adding device-driver modules to a driver map" on page 131

- "Working with driver-map settings and menu-item properties" on page 131

Creating a new driver map based on an existing driver map

This procedure describes how to create a new driver map using an existing driver map as a template. The new driver map will contain all of the module entries and settings that were contained in the source driver map. To create a new driver map based on an existing driver map, complete the following procedure:

1. Open the repository where you want the new driver map to reside.
2. If your repository contains folders, navigate to the folder where you want the new driver map to reside.
3. From the Main window menu bar, click **Insert** and then click **Driver Map...**. The New Map Wizard opens.
4. Click **Next**.
5. Verify that the **Driver Map** radio button is selected, and then click **Next**.
6. In the Name field, type the name you want to use to identify your new driver map. Be as specific as possible. This name will be used to identify your map in the repository.
7. Click **Next**.
8. Select the driver map you want to use as the source and then click **Next**.
9. Click **Finish**. The Driver Map window opens for the new driver map.
10. Make whatever changes are needed to the map.
11. When you have finished making your changes, click the **Save** icon located in the Main window toolbar; then, close the Driver Map window.

Adding menu items to a driver map

Menu items in a driver map provide a mechanism for annotation and the means by which filters are assigned to individual device-driver modules or groups of device-driver modules. Unlike base maps, menu items in a driver map have no effect on the menu items shown at the target computer.

To add a menu item to a driver map, complete the following procedure:

1. Open the repository where the driver map resides.
2. If your repository contains folders, navigate to the folder where the driver map resides.
3. In the right pane of the Repository window, click the **Driver Maps** tab. A list of driver maps is displayed.
4. Double-click the appropriate driver map. The Driver Map window opens.
5. If the map tree already has menu items in it, expand the map tree.
6. Click on the root entry of the tree structure
7. In the Main window menu bar, click **Insert** and then click **Menu Item**. A "New Item" box is added to the tree.
8. In the "New Item" box, type the text for the new menu item and then press Enter.
9. Repeat steps 6 through step 8 until all menu items are added.
10. To move a menu item up or down, click the menu item; then, click the **Move Item Up** or **Move Item Down** icon in the Main window toolbar.
11. When you are finished, click the **Save** icon in the Main Window.

Note: To delete modules or menu items within a map, select the module or menu item and click the **Delete** icon located in the Main window toolbar.

Adding device-driver modules to a driver map

To add one or more device-driver modules to a driver map, do the following:

1. Open the repository where the driver map resides.
2. If your repository contains folders, navigate to the folder where the driver map resides.
3. In the right pane of the Repository window, click the **Driver Maps** tab. A list of driver maps is displayed.
4. Double-click the appropriate driver map. The Driver Map window opens.
5. Expand the map tree.
6. Highlight either the root entry or the menu-item to which you want to add the device-driver module.
7. In the Main window menu bar, click **Insert** and then click **Device Driver...**. The Add Device Drivers window opens.
8. In the left pane of the Add Device Drivers window, navigate to the folder that contains the device-driver module you want to add.
9. In the right pane of the Add Device Driver window, click the module you want to add. If you want to add more than one module at a time, use the Shift key or Ctrl key in conjunction with the mouse to highlight multiple modules.
10. Click **OK**.
11. Repeat steps 6 through 10 for each device-driver module you want to add.
12. When you are finished, click the **Save** icon in the Main window toolbar.

Working with driver-map settings and menu-item properties

Driver-map settings apply to the entire map. You use the driver-map settings to add comments about the map, assign keywords to a map, or enable or disable the network-sync function. Menu-Item properties apply to a specific menu item. You use menu-item properties to change the text of a menu item or to associate one or more filters with a menu item. Unlike menu items in a base map, menu items in a driver map are not displayed as part of the menu system. The only purpose of driver-map menu items is for annotation and creating a structure to assign filters.

Specifying driver-map settings

The driver-map settings enables you to do the following:

- Add comments to a map
- Assign keywords to a map
- Enable or disable the network-sync function

To work with the driver map settings, do the following:

1. Open the repository where the driver map resides.
2. If your repository contains folders, navigate to the folder where the driver map resides.
3. In the right pane of the Repository window, click the **Driver Maps** tab. A list of driver maps is displayed.
4. Double-click the appropriate driver map. The Driver Map window opens.

5. In the Main window menu bar, click **Tools** and then click **Settings...**. The Map Settings window opens.
   - In the General tab, you can do the following:
     – If you want to add comments to the map, you can do so by typing free-form text in the Comments field. The Word Wrap check box enables you to automatically wrap text within the confines of the Comments field or establish your own line length and use a horizontal scroll bar.
     – If you want to add keywords to your map, you can do so by typing one or more keywords in the Keywords field. You can use any characters in a keyword except for the vertical bar (|) and the ampersand (&). Separate keywords with a space.

       Note: Keywords enable the developer to create a *Deployment Disc* (CD or DVD) that limits the selection of maps to only those that have one or more matching keywords. The ImageUltra Builder program automatically assigns some keywords to maps based on the certain map settings and the build status. You can manually assign other keywords to maps through the Map Settings window. For more information about predefined keywords and how keywords are used, refer to the ImageUltra Builder help system.
   - To set or disable the network-sync function, do the following in the Network tab:
     – Place a check mark in the **Use Network Sync** check box to enable the network-sync function. The network-sync function checks for updated maps during the deployment and installation processes.
     – Remove the check mark from the **Use Network Sync** check box to disable the network-sync function.
   - To enable or disable driver map filter processing, do the following in the Network tab:
     – Place a check mark in the **Use driver map filter processing** check box if you want only those device-driver modules that apply to the target computer to be deployed as part of your Smart Image.
     – Leave the **Use driver map filter processing** check box empty if you want all device-driver modules defined in your map to be deployed as part of your Smart Image.
6. When you are finished making changes, click **OK**; then, click the **Save** icon in the Main window toolbar.

Specifying menu-item properties in a driver map

Each menu item in a driver map, including the root entry, has a unique set of properties associated with it. The menu-item properties control the following:

- **General:** This property controls the text of the menu item or root entry. If you change the text of the root entry, you change the name of the map in the repository.
- **Conditions:** This property is available for menu items only; it is not available for the root entry. You use the Conditions property to assign one or more filters to a menu item.
- **Exclude:** This property is available for menu items only; it is not available for the root entry. You can use Exclude property to exclude specific device drivers in your map from being installed.

To work with the driver map settings, do the following:

1. Open the repository where the driver map resides.

2. If your repository contains folders, navigate to the folder where the driver map resides.
3. In the right pane of the Repository window, click the **Driver Maps** tab. A list of driver maps is displayed.
4. Double-click the appropriate driver map. The Driver Map window opens with the root entry highlighted.
5. Expand the map tree and select the appropriate entry (root entry or menu item).
6. In the Main window menu bar, click **View** and then click **Properties**. The Menu Item Properties window opens.
7. In the General tab, you can change the name of the selected menu item or the root entry by changing the text in the Description field.
8. Click the **Conditions** tab if you want to associate one or more filters with the selected menu item, then do the following:

   Note: Typically, filters are associated with menu items only when the modules listed under the menu item are tied to a certain machine condition. For example, if you are creating a driver-map tree structure that contains device drivers for multiple machine types, you can use a filter to determine the machine type of the target computer and install only those device-driver modules that apply to a specific machine type. When a filter is associated with a menu item in a driver map, the filter is automatically run during the installation process at the target computer.

   a. In the Conditions tab, click the **Insert Filter** icon. The Add Filters window opens.
   b. In the left pane of the Add Filters window, navigate to the folder that contains the filter you want to associate with the menu item
   c. In the right pane of the Add Filters window select the filter you want to associate with the menu item, then click **OK**. The Add filters window closes and the filter is added to the Conditions tab.
   d. Double-click the filter you just added. The Edit Parameters window opens.
   e. In the Parameters field, type any parameters required for the filter. For information about the parameters used with the filters provided as part of the ImageUltra Builder program, refer to the ImageUltra Builder help system.
   f. Repeat steps a through e for each filter you want to associate with the menu item.
   g. If you have more than one filter associated with a menu item, do one of the following:
      - If you want the menu item to be displayed only if the conditions set by *all* filters are met, place a check mark in the **All conditions must be satisfied** check box.
      - If you want the menu item displayed if the conditions set by *at least one* filter is met, do not place a check mark in the A**ll conditions must be satisfied** check box.
9. When you have finished making the changes, click the **Save** icon located in the Main window toolbar; then, close the Menu Item Properties window.

Chapter 8. Preparing source files for modules

This chapter provides information about preparing source files and creating modules from source files.

Creating source operating-system images

The method you use to create the source for a base operating-system module depends on the type of image you intend to deploy to the target computer. The methods for creating the source for a Portable-Sysprep Image, Hardware-Specific Image, and Ultra-Portable Image differ.

Ultra-Portable Image source files

The ImageUltra Builder program provides the tools necessary to do the following:

- *Import* an ImageUltra-developed base operating-system module from either an IBM branded personal computer or from a set of ThinkCentre and ThinkPad Product Recovery CDs for use with Ultra-Portable Images on IBM branded computers only. For more information about importing modules, see "Importing pre-existing maps and modules" on page 13 or consult the ImageUltra Builder help system for step-by-step instructions.

- *Create* an I386 base operating-system module and the associated UNATTEND.TXT module for use with Ultra-Portable Images for use on IBM branded and non-IBM branded computers.

  - For information about preparing the source files for an I386 base operating-system module and the associated modules, see "Preparing source files for an I386 base operating-system module."

  - For information about preparing the source files for the UNATTEND.TXT add-on operating-system module, see "Preparing source files for the UNATTEND.TXT module" on page 138.

  - For information about preparing the source files for the Sysprep module, see "Preparing the Sysprep module source file" on page 136.

  - For an overview of the I386 base operating-system module and other associated modules used to create an Ultra-Portable Image, see "Special considerations for I386 base operating-system modules" on page 84.

In addition to installing application modules, add-on operation-system modules, and device driver modules as part of an Ultra-Portable Image, you also can use *partitioning modules* to create an additional partition. By default, Ultra-Portable Images are limited to a single-partition installation. But by using partitioning modules, you can create an empty partition, create a partition that contains data, or delete one or more partitions. Any data that needs to be in the additional partition must be contained in the partitioning module; you cannot install other modules in a data partition created by a partitioning module. Instructions for preparing files for the various types of modules is provided later in this chapter. For information about incorporating partitioning modules in a map, see "Using a partitioning module in a base map" on page 96.

Preparing source files for an I386 base operating-system module

The source files for an I386 base operating-system module are available in the I386 folder of a Windows XP Home, Windows XP Professional, or Windows 2000

installation CD. These files are operating-system specific as well as language specific. Therefore, you must create separate source folders for each operating system/language combination.

The best way to prepare these source files is to create separate folders for each operating-system/language combination on your local drive or on a network drive. If you are using a network drive, make sure the ImageUltra Builder console can access the source files. The following is an example of a folder structure for the source files:

```
X:\ULTRAPORTABLE\XP_PRO\FRENCH
X:\ULTRAPORTABLE\XP_HOME\FRENCH
X:\ULTRAPORTABLE\2000\FRENCH
```

After you create the folder structure, copy the complete I386 folder from the appropriate Windows installation CD to the folder you created for it. The following is an example of the folder structures after copying the I386 folder from the various Windows installation CDs:

```
X:\ULTRAPORTABLE\XP_PRO\FRENCH\I386
X:\ULTRAPORTABLE\XP_HOME\FRENCH\I386
X:\ULTRAPORTABLE\2000\FRENCH\I386
```

Once you have copied the source files from the Windows installation CD, remove the read-only attribute and rename the UNATTEND.TXT file.

Note: Although one file in the I386 folder, the UNATTEND.TXT file, contains customer-unique product identification (PID) and locale information, the ImageUltra Builder program enables you to create this file externally and replace the file dynamically during the installation process. The external UNATTEND.TXT file enables you to maintain a single set of I386 source folders and modules that can be used for a variety of customers and locales. For more information about preparing the UNATTEND.TXT file, see "Setting the attributes for an UNATTEND.TXT module" on page 139.

Continue with "Setting the attributes for an I386 base operating-system module" on page 137.

Preparing the Sysprep module source file
To prepare the Sysprep module source file, do the following:
1. Obtain the following sysprep files: SYSPREP.EXE, SETUPCL.EXE, and FACTORY.EXE (for Windows XP only). For more information about obtaining the sysprep files, refer to "Creating and preparing the image files" on page 143.
2. Copy the files into a folder structure. For example: X:\ultraportable\xp\sysprep.

Creating the Sysprep module source file
To create the sysprep module, do the following:
1. Create a new Add-on operating system module.
2. Update the description and version fields (if desired). Update the language and OS fields.
3. In the Source tab, change the source directory to X:\ultraportable\xp\sysprep.
4. In the Options tab, change the "Path to copy installable files" to: \IBMWORK\SYSPREP.

5. Click **Save**, and then build the module.

After you build the module copy it into the unattended install map with the Primary pratition, the I386 module, and the UNATTEND.TXT module.

Setting the attributes for an I386 base operating-system module

The following information describes how to set the attributes for a new I386 base operating-system module.

To set the attributes for the new module, do the following:

1. Open the repository where you want the new base operating-system module to reside.
2. If your repository contains folders, navigate to the folder where you want the new I386 base operating-system module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Operating System** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

   Note: It is a good idea to include the image type as part of the name, such as (UPI) for Ultra-Portable Image. This will help you when you create your maps.

7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. Click **Next** without selecting any of the modules listed.
10. Click the **I386 base operating-system module** radio button.
11. Click **Next**.
12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.
13. In the General tab, do the following:

    a. In the Version field, type the full version number of the operating system.

    b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

    c. In the Comments field, you can remove any default comments and replace them with any comments that you want to keep with the module.

14. In the OS/Languages tab, do the following:

    a. In the left pane, put a check mark next to each language for which the module is intended to be used. In most cases, I386 base operating-system modules are language specific and only one language is typically selected. However, you can select multiple languages, if needed.

b. In the right pane, put a check mark next to the operating system for which this module applies.
15. In the Source tab, do the following:
   a. In the Source Directory field, type the complete path to the folder that contains the I386 source files.

      Note: Do not use the I386 folder created during a previous installation. These files must come the I386 folder of a Windows installation CD or files copied to another folder directly from a Windows installation CD.

   b. Leave the remaining fields blank.
16. Do not make any modifications in the Filters or Options tab.
17. Click the **Save** icon in the Main window tool bar.
18. Your I386 base operating-system module is ready to be built. Continue with "Building a new module" on page 166.

Preparing source files for the UNATTEND.TXT module

Each I386 base operating-system module requires at least one UNATTEND.TXT add-on operating-system module. The UNATTEND.TXT module contains one file:

- A customized UNATTEND.TXT file (either UNATTEND.W2K or UNATTEND.WXP, depending on the operating system you are installing). This UNATTEND.TXT file enables the operating system to be installed either unattended or moderately-attended.

The UNATTEND.W2K and UNATTEND.WXP files are provided as part of the ImageUltra Builder program in the \SOURCE\COPYI386 folder of your repository. You cannot access these source files from the ImageUltra Builder interface; you must use Windows Explorer or My Computer. If you cannot remember the path to your repository, in the ImageUltra Builder Main window, click **File** and then click **Open Repository...**. The repository paths are listed. You can use any repository that has been created by or updated by the ImageUltra Builder 3.0 program.

Preparation of these source files involves the following steps:

1. Set up a source folder for each operating-system/language/customer combination. For example:

`X:\ULTRAPORTABLE\2000\FRENCH\LENOVO\UNATTEND`

2. Copy the proper UNATTEND file of the \SOURCE\COPYI386 folder (from your repository) into each source folder. Rename it to *.TXT. This yields the following file in the source folder:

`X:\ULTRAPORTABLE\2000\FRENCH\LENOVO\UNATTEND\UNATTEND.TXT`

3. Modify the UNATTEND.TXT file. At a minimum, you must change the following:
   - The value in the ProductID field for Windows 2000
   - The value in the ProductKey field for Windows XP

   Note: For more information about the fields in the UNATTEND.TXT file, see the following resources:
   - The Microsoft Web site at `http://www.microsoft.com`. Search for the string `UNATTEND.TXT`.
   - Your Windows 2000 installation CD. Locate the UNATTEND.DOC file.

- Your Windows XP installation CD. Open the DEPLOY.CHM file located in the DEPLOY.CAB file.

Continue with "Setting the attributes for an UNATTEND.TXT module."

Setting the attributes for an UNATTEND.TXT module

The following information describes how to set the attributes for a new UNATTEND.TXT add-on operating-system module.

To set the attributes for the new module, do the following:

1. Open the repository where you want the new base operating-system module to reside.
2. If your repository contains folders, navigate to the folder where you want the new UNATTEND.TXT add-on operating-system module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Operating System** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

   Note: It is a good idea to include UNATTEND.TXT as part of the name, such as UNATTEND.TXT FOR XPHOME FRENCH. This will help you when you create your maps.
7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. Click **Next** without selecting any of the modules listed.
10. Click the **Add-on operating system** radio button.
11. Click **Next**.
12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.
13. In the General tab, do the following:
    a. In the Version field, you can type a version number if needed.
    b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.
    c. In the Comments field, type any comments that you want to keep with the module.
14. In the OS/Languages tab, do the following:
    a. In the left pane, put a check mark next to each language for which this UNATTEND.TXT module is intended to be used. In most cases, UNATTEND.TXT modules are language specific and only one language is typically selected. However, you can select multiple languages, if needed.

b. In the right pane, put a check mark next to the operating system for which this UNATTEND.TXT module applies.

15. In the Source tab, do the following:

   a. In the Source Directory field, type the complete path to the folder that contains the appropriate UNATTEND.TXT source file.

   b. Leave the remaining fields blank.

16. Do not make any modifications in the Filters tab.

17. Click the **Save** icon in the Main window tool bar.

18. In the Options tab, change the "Path to copy installable files" to \I386 and the "Target partition" to C:.

19. Your UNATTEND.TXT add-on operating-system module is ready to be built. Continue with "Building a new module" on page 166.

Required third-party tools

If you intend to create either Portable-Sysprep Images or Hardware-Specific Images, you will need to use the Windows Sysprep tool and do the following:

- Understand the requirements for running the Sysprep tool
- Familiarize yourself with the several languages that the Sysprep tool supports
- Understand which versions of the Sysprep tool are compatible with the operating systems you will use for your images
- Familiarize yourself with at least one of the supported third-party image-cloning tools: Symantec Norton Ghost or PowerQuest DeployCenter

For more information about using the Sysprep tool, go to the Microsoft Windows Web site at http://www.microsoft.com/ms.htm and search on the term Sysprep.

For more information about Symantec Norton Ghost, go to the Symantec Web site at http://www.symantec.com/ search on the term Ghost.

For more information about PowerQuest DeployCenter, go to the PowerQuest Web site at http://www.powerquest.com/ and search on the term DeployCenter.

Limitations of Symantec Norton Ghost

The following limitations must be considered when Symantec Norton Ghost is used in conjunction with this version of the ImageUltra Builder program:

- You can only use Win32 programs to restore your custom portable sysprep or hardware specific images.
- Currently, the only version of Symantec Norton Ghost that includes a 32-bit version of Ghost that is compatible with Windows PE is Ghost Corporate 8 Edition. This version of Ghost contains the GHOST32.EXE file which is compatible with ImageUltra and Windows PE.

Cloned drives and partitions: Symantec Norton Ghost can clone partitions and drives. Because the ImageUltra Builder program uses a service partition to store the Smart Image for certain types of installations, Symantec Norton Ghost might encounter problems when attempting to set the size for multiple partitions or when installing a cloned drive. When defining partition sizes as a percentage, Ghost always uses the full hard-disk capacity and does not take the presence of a service partition into account. As a result, Ghost might delete the service partition on the target computer. Defining absolute sizes can be equally problematic because drive sizes vary on target computers.

When using a cloned drive, two problems can occur:

- If a service partition exists on a source computer, Ghost includes this service partition as part of the image and will overwrite the service partition on the target computer. This results in a mismatch between the contents on the active partition of the target computer and the service partition of the target computer.
- If a service partition does not exist on the source computer, Ghost will assume that it can use the entire hard disk for the image, and will delete the service partition on the target computer.

Because it is so difficult to manage multiple partitions and cloned drives created by Ghost when a service partition is present, support is provided for the following implementations only:

- Restoring single-partition images produced by Ghost
- Restoring a single-partition, cloned-drive image as a single partition, not as a cloned drive.

The Silent Install command in the Source tab of the Operating System window must be a batch file name, such as GHOST.BAT.

Inside the batch file, you must use the following statements:

```
ECHO SELECT DISK 0 >%RAMD%\S.SCP
ECHO CREATE PARTITION PRIMARY >>%RAMD%\S.SCP
ECHO EXIT >>%RAMD%\S.SCP
DISKPART /s %RAMD%\S.SCP
ghost32.exe -clone,mode=pload,src=image.gho:1,dst=1:1 -auto -sure -quiet -fx -batch
```

Where *image.gho* is the name of your image.

Before you build the module, you must put the image files, the script file, and the batch file in the same folder.

File size and naming convention: It is always a good idea to limit the file size to 600MB so the image can be distributed on a set of CD's, if necessary. If you are using Symantec Norton Ghost version 7.5, use the -cns parameter when creating the image to ensure the file-extension names are compatible with the ImageUltra Builder program.

Limitations of PowerQuest DeployCenter

The ImageUltra Builder program uses a service partition at the end of the hard disk to store the Smart Images. The service partition is hidden and must be protected during the installation process. PowerQuest DeployCenter has a scripting function that allows for the protection of partitions.

Before any files are installed, the ImageUltra Builder program deletes all partitions from the target computer hard disk except for the service partition. Therefore, when creating scripts, you do not have to include any commands to delete partitions.

The following is a sample PowerQuest DeployCenter script that can be used with the ImageUltra Builder program to protect the service partition and restore two partitions:

```
SELECT DRIVE 1
PROTECT PARTITION LAST
```

```
SELECT FREESPACE FIRST
SELECT IMAGE 1
RESIZE IMAGE PROPORTIONAL
SELECT IMAGE 2
RESIZE IMAGE 2048
RESTORE
SELECT PARTITION 1
SET ACTIVE
SELECT PARTITION 2
UNHIDE
```

In the sample script, two partitions are established: partition 1 uses all available free space and partition 2 uses 2 GB. By default, when PowerQuest DeployCenter sets partition 1 as active, it hides any remaining primary partitions. Therefore, the script must "unhide" any primary partitions that were hidden when partition 1 was set as active. The sample script "unhides" partition 2. If a third primary partition were being used, you would add the following lines to the end of the script:

```
SELECT DRIVE 1
SELECT PARTITION 3
UNHIDE
```

When building a module for a PowerQuest image, use a batch file and use the batch file name as the Silent Install command in the Source tab of the Operating System window. For example, use PQIMAGE.BAT as the Silent Install command.

Inside the PQIMAGE.BAT file, you must use the following line:

```
pqideploy.exe /cmd=restore.scp /I24 /NRB /img=image.pqi
```

where *restore.scp* is the script shown earlier and *image.pqi* is the name of the image that you want to restore. The /NRB parameter instructs the PQDI.EXE program not to reboot the computer so the ImageUltra Builder installation program can regain control after the PowerQuest image has been installed.

Before you build the module, you must put the image files, the script file, and the batch file in the same folder.

For additional information about developing PowerQuest DeployCenter scripts, refer to your PowerQuest DeployCenter documentation.

Creating a Portable-Sysprep Image

There are four main steps to creating a base operating-system module based on a Portable-Sysprep Image:

1. Making the ImageUltra Customization Program accessible to the source computer
2. Creating the Portable-Sysprep Image on a source computer and using Symantec Norton Ghost or PowerQuest DeployCenter to create the image files
3. Setting the module attributes
4. Building the module from the image files

Making the ImageUltra Customization Program accessible

Before you can create a Portable-Sysprep Image, you must first make the ImageUltra Customization Program (IUBCP.EXE) accessible to the source computer.

The ImageUltra Builder program enables you to copy the ImageUltra Customization Program on a CD, shared drive, or any other media to which the ImageUltra console has direct-write access.

1. Use Windows Explorer or My Computer to open the IUBCP folder in your repository.

   Note: If you cannot remember the path to your repository, in the ImageUltra Builder Main window, click **File** and then click **Open repository...**. The repository paths will be listed. You can use any repository.

2. Copy the IUBCP.EXE program to a CD or to a shared drive that the source computer can access during the development of the image. If you use a CD, label the CD "ImageUltra Customization Program."

3. Do one of the following:
   - Provide an *ImageUltra Customization Program* CD to any personnel responsible for creating Portable-Sysprep Images.
   - Provide the path to the IUBCP.EXE program to any personnel responsible for creating Portable-Sysprep Images.

Creating and preparing the image files

Before you begin, make sure you put the ImageUltra Customization Program either on portable media that is compatible with the source computer or a shared drive that the source computer is able to access after the operating system and other programs have been installed. See "Making the ImageUltra Customization Program accessible" on page 142 for more information.

To create and prepare Portable-Sysprep Image source files, do the following:

1. Install the Windows operating system (Windows XP or Windows 2000) that you want on your target computers.

   Note: It is a good idea to use your image-cloning tool to take a snapshot of the image as it exists at this point (*before* you run the ImageUltra Customization Program) in case you need to make changes to the image. If you do not take a snapshot at this point, you will have to recreate your image beginning from step 1 if changes are needed.

2. Access and run the ImageUltra Customization Program (IUBCP.EXE). This program might be on portable media or a shared drive. See "Making the ImageUltra Customization Program accessible" on page 142 for details.

3. Obtain the Microsoft Sysprep program designed for your operating system:
   - The Windows 2000 version of the Sysprep program is available from the Microsoft Web site at
     `http://www.microsoft.com/windows2000/downloads/tools`
     `/sysprep/license.asp`.
     You will need the following files: SYSPREP.EXE and SETUPCL.EXE.
   - The Windows XP version of the Sysprep program is on the Windows XP CD in the \SUPPORT\TOOLS\DEPLOY.CAB file. You will need the following files: SYSPREP.EXE, SETUPCL.EXE, and FACTORY.EXE.

4. Copy the Sysprep files to the source computer as follows:
   - Windows 2000: Copy the SYSPREP.EXE and SETUPCL.EXE files into the C:\IBMWORK\SYSPREP folder and C:\SYSPREP folder.
   - Windows XP: Copy the SYSPREP.EXE, SETUPCL.EXE, and FACTORY.EXE files into the C:\SYSPREP folder.

5. The ImageUltra Customization Program creates an empty TAG file for Windows XP Home, Media Center, Professional, and Tablet PC as follows:

- Windows 2000 (C:\IBMWORK\WIN2000)
- Windows XP Professional (C:\IBMWORK\XPPRO)

   Note: If you are making a Windows XP Home image, rename
   C:\IBMWORK\XPPRO to C:\IBMWORK\XPHOME.

6. Install any application programs that will be common across all of your target computers. If you have any applications that you cannot make into a module, now is the time to install them.

   Note: This step is optional. You can include common applications as part of your operating-system module or you can build separate application modules and control them through your base map.

7. Make any other modifications to the image that you feel are necessary.

8. Make any required changes to the C:\IBMWORK\SYSPREP.INF file for your image. This file is used to re-enter the audit-boot mode. You might change a password, set PID, and so on.

   Note: You might have to update PID information. In Windows XP, the easiest place to modify the PID information is in C:\IBMWORK\PROKEYP.FM.

9. Make any required changes to customize Sysprep in the C:\SYSPREP\SYSPREP.INF file.

10. Run SYSPREP.EXE as follows:
    - Windows 2000: Run C:\SYSPREP\SYSPREP -quiet
    - Windows XP: Run C:\SYSPREP\SYSPREP.EXE -factory -quiet

    Note: If you are making a Windows XP Home image, make sure you are logged in as the *Owner* account before running the above sysprep command.

11. **Important:** Before you continue to the next step, read "Limitations of Symantec Norton Ghost" on page 140 or "Limitations of PowerQuest DeployCenter" on page 141 to learn about special preparation procedures associated with your image-cloning tool.

12. Use either Symantec Norton Ghost or PowerQuest DeployCenter software to create the image files. If you intend to deploy the image using CDs, make sure you use the tools provided with Ghost and DeployCenter to segment the image into files that can fit on CDs.

    Note: Set the maximum file size for CDs at 600MB. If you are using Symantec Norton Ghost, make sure you use the *-cns* parameter when creating the image.

13. Store the image file in its own folder. In the case of CD deployment, where you had to segment the image into CD-sized files, store all of the files in a single folder. Make sure the folder is in a location that is accessible to the ImageUltra Builder console.

14. Store the Image-cloning tool you used to create the image (Symantec Norton Ghost or PowerQuest DeployCenter) in the same folder as the image file.

15. Store any batch files or script files you may have created in the same folder as the image file.

16. Continue with "Setting the attributes for the new base operating-system module" on page 145.

Setting the attributes for the new base operating-system module

The following information describes how to set the attributes for a new Portable-Sysprep Image base operating-system module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing base operating-system module" on page 146.

To set the attributes for the new module, do the following:

 1. Open the repository where you want the new base operating-system module to reside.
 2. If your repository contains folders, navigate to the folder where you want the new base operating-system module to reside.
 3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.
 4. Click **Next**.
 5. Verify that the **Operating System** radio button is selected, and then click **Next**.
 6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

    Note: It is a good idea to include the image type as part of the name, such as (PSI) for Portable-Sysprep Image. This will help you when you create your maps.

 7. Click the **No** radio button to indicate that this module is not a container.
 8. Click **Next**.
 9. Click **Next** without selecting any of the modules listed.
 10. Click the **Base Operating System** radio button, then use the drop-down menu to select **Portable-Sysprep Image**.
 11. Click **Next**.
 12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.
 13. In the General tab, do the following:

    a. In the Version field, type the full version number of the operating system.

    b. In the Type field, verify that **Portable-Sysprep Image** is selected.

    c. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

    d. In the Comments field, type any comments that you want to keep with the module.

 14. In the OS/Languages tab, do the following:

    a. In the left pane, put a check mark next to each language for which the module is intended to be used. In most cases, base operating-system modules are language specific and only one language is typically selected. However, you can select multiple languages, if needed.

b. In the right pane, put a check mark next to the operating system for which this module applies.

15. In the Source tab, do the following:

a. In the Source Directory field, type the complete path to the folder that contains the image files.

Note: Make sure that this folder does not contain any unnecessary files.

b. In the Silent Install Command area, type the name of the batch file you developed to control the installation of your image in the Command field, and if parameters are required for the batch file, type the parameters in the Parameters field. For more information about creating this batch file, see "Limitations of Symantec Norton Ghost" on page 140 and "Limitations of PowerQuest DeployCenter" on page 141.

16. Click the **Save** icon in the Main window tool bar.

17. Continue with "Building a new module" on page 166.

Setting the attributes based on an existing base operating-system module

To set the attributes for a new Portable-Sysprep Image base operating-system module to match the attributes of an existing base operating-system module, do the following:

1. Open the repository where you want the new base operating-system module to reside.

2. If your repository contains folders, navigate to the folder where you want the new base operating-system module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Operating System** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. From the list of modules, select the module you want to use as the source and then click **Next**.

10. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.

11. Make any necessary changes. Refer to the ImageUltra Builder help system if you need assistance.

12. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar.

13. Continue with "Building a new module" on page 166.

Creating a Hardware-Specific Image

There are three main steps to creating a base operating-system module based on a Hardware-Specific Image:

1. Creating the Hardware-Specific Image on a source computer and using Symantec Norton Ghost or PowerQuest DeployCenter to create the image files

2. Setting the module attributes

3. Building the module from the image files

Creating and preparing the image files

To create and prepare Hardware-Specific Image source files, do the following:

1. On a source computer, install the Windows operating system that you want on your target computers. You can use any file system that you want.

2. Install any application programs that will be common across all of your target computers.

3. Install the Microsoft Sysprep program designed for your operating system in the C:\SYSPREP folder. Do not run SYSPREP.EXE at this time.

   Note: The Windows 2000 version of the Sysprep program is available from the Microsoft Web site at `http://www.microsoft.com/windows2000/download/tools /sysprep/license.asp`. The Windows XP version of the Sysprep program is on the Windows XP CD in the \SUPPORT\TOOLS\DEPLOY.CAB file.

4. Make any required changes to the C:\SYSPREP\SYSPREP.INF file and then run SYSPREP.EXE.

5. In most cases the Sysprep program shuts down the computer when it has finished running. If the Sysprep program does not shut down the source computer, shut down the source computer manually.

   Important: Before you continue to the next step, read "Limitations of Symantec Norton Ghost" on page 140 or "Limitations of PowerQuest DeployCenter" on page 141 to learn about special preparation procedures associated with your image-cloning tool.

6. Use either Symantec Norton Ghost or PowerQuest DeployCenter software to create the image. If you intend to deploy the image using CDs, make sure you use the tools provided with your image-cloning tool to segment the image into files that can fit on CDs.

   Note: Set the maximum file size for CDs at 600MB. If you are using Symantec Norton Ghost, make sure you use the -cns parameter when creating the image.

7. Store the image in its own folder. In the case of CD deployment, where you had to segment the image into CD-sized files, store all of the files in a single folder. Make sure the folder is in a location that is accessible by the ImageUltra Builder console.

8. Store the image-cloning tool that you used to create the image (Symantec Norton Ghost or PowerQuest DeployCenter) in the same folder as the image file.

9. Store any batch files or script files that you may have created in the same folder as the image file.

10. Continue with "Setting the attributes for the new base operating-system module."

Setting the attributes for the new base operating-system module

The following information describes how to set the attributes for a new Hardware-Specific Image base operating-system module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing base operating-system module" on page 146.

To set the attributes for the new module, do the following:

1. Open the repository where you want the new base operating-system module to reside.
2. If your repository contains folders, navigate to the folder where you want the new base operating-system module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Operating System** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

   Note: It is a good idea to include the image type as part of the name, such as (HSI) for Hardware-Specific Image. This will help you when you create your maps.

7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. Click **Next** without selecting any of the modules listed.
10. Click the **Base Operating System** radio button, then use the drop-down menu to select **Hardware-Specific Image**.
11. Click **Next**.
12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.
13. In the General tab, do the following:
    a. In the Version field, type the full version number of the operating system.
    b. In the Type field, verify that **Hardware-Specific Image** is selected.
    c. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

    d. In the Comments field, type any comments that you want to keep with the module.
14. In the OS/Languages tab, do the following:
    a. In the left pane, put a check mark next to each language for which the module is intended to be used. In most cases, base operating-system modules are language specific and only one language is typically selected. However, you can select multiple languages, if needed.
    b. In the right pane, put a check mark next to the operating system for which this module applies.
15. In the Source tab, do the following:
    a. In the Source Directory field, type the complete path to the folder that contains the image files.
    b. In the Silent Install Command area, type the name of the batch file you developed to control the installation of your image in the Command field,

and if parameters are required for the batch file, type the parameters in the Parameters field. (See "Limitations of Symantec Norton Ghost" on page 140 and "Limitations of PowerQuest DeployCenter" on page 141 for information about creating this batch file.)

16. Click the **Save** icon in the Main window tool bar.

17. Continue with "Building a new module" on page 166.

Setting the attributes based on an existing base operating-system module

To set the attributes for a new Hardware-Specific Image base operating-system module to the match the attributes of an existing base operating-system module, do the following:

1. Open the repository where you want the new base operating-system module to reside.

2. If your repository contains folders, navigate to the folder where you want the new base operating-system module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Operating System** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. From the list of modules, select the module you want to use as the source and then click **Next**.

10. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.

11. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.

12. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar.

13. Continue with "Building a new module" on page 166.

Preparing source files for application modules and add-on operating-system modules

Any application or add-on operating-system component that you intend to deploy as a module must be prepared to perform a silent installation.

Note: The exception is the UNATTEND.TXT add-on operating-system module, which must be used in conjunction with an I386 base operating-system module. For instructions on preparing source files for an UNATTEND.TXT add-on operating-system module, see "Preparing source files for the UNATTEND.TXT module" on page 138.

In addition, you can develop and include mini-applications that run immediately before or after an application or add-on operating-system component installs. Because the method of preparing applications and add-on operating-system components are nearly identical, the term application will be used throughout this

section to describe both. The terms *add-on operating-system component* and *add-on operating-system module* will be used when these items need to be distinguished from an *application* or *application module*.

Preparing for a silent install

The method you use to prepare an application for silent installation varies depending on the installation mechanism being used. For example, if the application uses InstallShield as the installation mechanism, the application might already be enabled for silent installation if a SETUP.ISS file exists in the same folder as the SETUP.EXE file. If a SETUP.ISS file does not exist, or if the provided SETUP.ISS file does not produce the desired result, InstallShield enables you to create a new SETUP.ISS file by doing the following:

Note: Perform the following steps on a test computer.

1. Copy all source files for the application into an empty folder. Make sure you maintain the folder structure that was provided on the software manufacturer's distribution media. In the case of a .ZIP file, make sure you maintain the folder structure when you unzip the file.
2. Open a command prompt window.
3. Change to the folder that contains the application SETUP.EXE file.
4. At the command prompt, type **SETUP -R** and then press Enter.
5. Follow the steps to install the program. As you install the program, the keystrokes and mouse clicks are recorded in the new SETUP.ISS file. Because the values you type will be used on all target computers, you should use company-related information in place of any personal information for which you might be prompted.
6. When the installation is complete, the new SETUP.ISS file will be in the Windows folder. Copy the new SETUP.ISS file from the Windows folder into the folder that contains the application SETUP.EXE file.

Note: New versions of InstallShield might have different steps.

Most commercially-available installation products, such as Microsoft Installer, also enable you to do a silent install. InstallShield Developer enables you to create silent installations for both the Basic MSI and InstallScript MSI project types.

To run a Basic MSI setup silently, you need to use the following command:
```
MSIEXEC /i Product.msi /qn
```

If your release settings include SETUP.EXE, you can run the following command:
```
SETUP.EXE /s /v"/qn"
```

Basic MSI projects do not create or read response files. To set installation properties for a Basic MSI project, use a command similar to the following:

```
MSIEXEC /i Product.msi /qn INSTALLDIR=D:\ProductFolder USERNAME='Valued Customer"
```

See the documentation associated with the specific installation product if you encounter any problems.

Adding mini-applications

The use of mini-applications is optional. Mini-applications are useful for creating special folders, cleaning up registry entries, or performing any other service related

to the application being installed. The content and format of a mini-application is entirely up to you. You are limited to running one mini-application before the application is installed and one mini-application after the application is installed.

The only requirement related to the preparation of mini-applications is that you must copy all of the mini-application files into the same folder as the application SETUP.EXE file or into one of its subfolders.

Making files accessible to the ImageUltra Builder console

The last step of preparation is making the files accessible to the ImageUltra Builder console. Simply copy the folder and subfolders containing the application source files to any drive that can be accessed by the ImageUltra Builder console. You must keep the folder structure intact.

After the files are prepared

After the files are prepared, you can create a module entry in the repository if one does not exist, and then build the module. When you create the module entry, you are given the opportunity to define a number of attributes associated with the module. The attributes that are directly related to the preparation steps described in this section are on the Source tab. Those attributes include:

- The source directory of the application
- The silent-install command and parameters
- The preinstall command and parameters (for a mini-application)
- The postinstall command and parameters (for a mini-application)

For additional information about setting the attributes, see one of the following:
- "Setting the attributes for a new application module"
- "Setting the attributes for a new add-on operating-system module" on page 154

Setting the attributes for a new application module

The following information describes how to set the attributes for a new application module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing application module" on page 153.

To set the attributes for the new module, do the following:

1. Open the repository where you want the new application module to reside.
2. If your repository contains folders, navigate to the folder where you want the new application module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Application...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Applications** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.
7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. Click **Next** without selecting any of the modules listed.

10. Click **Finish**. The New Module Wizard closes and the Application window for the new module opens.

11. In the General tab, do the following:

    a. In the Version field, type the full version of the application.

    b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

    c. In the Comments field, type any comments you want kept with the module.

12. In the OS/Languages tab, do the following:

    a. In the left pane, put a check mark next to each language for which the application will be used. For example, you might choose to use an English application module with a different language operating-system module.

    b. In the right pane, put a check mark next to the operating system for which this application module is intended to be used. If the application is not specific to any Windows operating system, you can check all of the boxes. If the application is specific to an operating system, check only the operating system that applies.

13. In the Source tab, do the following:

    a. In the Source Directory field, type the complete path to the folder that contains the application files.

       Note: In the following step, the fields in the Silent Install Command area are used for applications that are to be installed through a normal installation process. If you want to copy the installable files to the target hard disk instead of installing the application, leave the fields in the Silent Install Command area blank, and fill in the "Path to copy installable files" field in the Options tab.

    b. In the Silent Install Command area, type the command that is to be used to install the application in the Command field and any associated parameters in the Parameters field. For example, your command might be SETUP.EXE and your parameters might be /S.

       Note: The following steps apply to the use of mini-applications. Mini-applications are useful for creating special folders, cleaning up registry entries, or performing any other service related to the installation of the application. For more information about the use of mini-applications, see "Preparing source files for application modules and add-on operating-system modules" on page 149.

    c. If you have a mini-application contained in this module that needs to run immediately *before* the actual application is installed, use the Preinstall Command area to type the command and parameters associated with starting the mini-application.

    d. If you have a mini-application contained in this module that needs to run immediately *after* the actual application is installed, use the Postinstall Command area to type the command and parameters associated with starting the mini-application.

14. If you want to assign a filter to the application module, do the following in the Filters tab:

    a. Click the **Insert Filter** icon. The Add Filter window opens.

    b. In the left pane of the Add Filters window, navigate to the folder that contains the filter you want to add.

    c. In the right pane of the Add Filter Window, select filter you want to use; then click **OK**. The Add Filter window closes and the filter is added to the Filters tab.

    d. Double-click on the filter you just added. The Edit Parameters window opens.

    e. In the Parameters field, type any parameters required to get the desired result. For information about the parameters associated with the filters provided with the ImageUltra Builder program, see the ImageUltra Builder help system.

    f. Click **OK**.

    g. If you assign multiple filters to a module, do one of the following:

       • If *all* filters must be satisfied to install the module, click the **All of these must be true** radio button.

       • If *any* of the filters can be satisfied to install the application, click the **One of these must be true** radio button.

15. In the Options tab, do the following:

    a. The **Path to copy installable files** field defines whether the application files should be copied to the C partition in an installable format.

       • If you type a path in this field and you provide a Silent Install Command in the Source tab, the files will be copied to the C partition in addition to the application being installed.

       • If you type a path in this field and you do not provide a Silent Install Command in the Source tab, the files will be copied to the C partition, but the application will not be installed automatically.

       Type a path only if you want the installable files to be copied to the C partition of the target computer.

    b. All other fields on the Options tab are used to handle special cases.

       • For information about the Install Hook and Install Slot fields, see Chapter 9, "Specifying the module-installation order," on page 167.

       • For information about specific fields on this tab, refer to the ImageUltra help system.

       • For other information about using this tab for special cases, see Chapter 11, "Dealing with exceptions," on page 175.

16. Click the **Save** icon in the Main window tool bar; then, close the Application window.

17. Continue with "Building a new module" on page 166.

Setting the attributes based on an existing application module

To set the attributes for a new application module to the match the attributes of an existing application module, do the following:

1. Open the repository where you want the new application module to reside.

2. If your repository contains folders, navigate to the folder where you want the new application module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Application...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Application** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. From the list of modules, select the module you want to use as the source and then click **Next**.

10. Click **Finish**. The New Module Wizard closes and the Application window for the new module opens.

11. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.

12. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar; then, close the Application window.

13. Continue with "Building a new module" on page 166.

Setting the attributes for a new add-on operating-system module

The following information describes how to set the attributes for a new add-on operating-system module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing add-on operating-system module" on page 156.

Note: If you are creating a an UNATTEND.TXT add-on operating-system module follow the instructions in "Setting the attributes for an UNATTEND.TXT module" on page 139.

To set the attributes for a new add-on operating-system module, do the following:

1. Open the repository where you want the new add-on operating-system module to reside.

2. If your repository contains folders, navigate to the folder where you want the new add-on operating-system module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Operating System** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. Click **Next** without selecting any of the modules listed.

10. Click the **Add on operating system module** radio button.

11. Click **Next**.

12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.

13. In the General tab, do the following:
    a. In the Version field, type the full version of the add-on component.
    b. If you want to assign a password to this module, type the encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.
    c. In the Comments field, type any comments you want kept with the module.
14. In the OS/Languages tab, do the following:
    a. In the left pane, put a check mark next to each language for which the add-on component will be used. For example, you might choose to use an English add-on operating-system component with multiple languages, or you might want to limit its use to English only.
    b. In the right pane, put a check mark next to the operating system for which this module applies. For example, if you are creating a module for the latest version of Windows Media Player, that module might apply to more than one operating system If you are creating a module for a service pack, that module would apply to one operating system only.
15. In the Source tab, do the following:
    a. In the Source Directory field, type the complete path to the folder that contains the source files for the add-on operating-system component.
    b. In the Silent Install Command area, type the command that is to be used to install the add-on operating-system component in the Command field and any associated parameters in the Parameters field. For example, your command might be SETUP.EXE and your parameters might be /S.

       Note: The following steps apply to the use of mini-applications. Mini-applications are useful for creating special folders, cleaning up registry entries, or performing any other service related to the installation of the add-on operating-system component. For more information about the use of mini-applications, see "Preparing source files for application modules and add-on operating-system modules" on page 149.
    c. If you have a mini-application contained in this module that needs to run immediately *before* the actual add-on operating-system component is installed, use the Preinstall Command area to type the command and parameters associated with starting the mini-application.
    d. If you have a mini-application contained in this module that needs to run immediately *after* the actual add-on operating-system component is installed, use the Postinstall Command area to type the command and parameters associated with starting the mini-application.
16. If you want to assign a filter to the add-on operating-system module, do the following in the Filters tab:
    a. Click the **Insert Filter** icon. The Add Filter window opens.
    b. In the left pane of the Add Filter window, navigate to the folder that contains the filter module you want to add.

c. In the right pane of the Add Filter window, select the filter you want to use and then click **OK**. The Add Filter window closes and the filter is added to the Filters tab.

d. Double-click the filter you just added. The Edit Parameters window opens.

e. In the Parameters field, type any parameters required to get the desired result. For information about the parameters associated with the filters provided with the ImageUltra Builder program, see the ImageUltra Builder help system.

f. Click **OK**.

g. If you assign multiple filters to a module, do one of the following:
   - If *all* filters must be satisfied to install the module, click the **All of these must be true** radio button.
   - If *any* of the filters can be satisfied to install the application, click the **One of these must be true** radio button.

17. In the Options tab, do the following:

a. The **Path to copy installable files** field defines whether the application files should be copied to the C partition in an installable format.
   - If you type a path in this field and you provide a Silent Install Command in the Source tab, the files will be copied to the C partition in addition to the application being installed.
   - If you type a path in this field and you do not provide a Silent Install Command in the Source tab, the files will be copied to the C partition, but the application will not be installed automatically.

   Type a path only if you want the installable files to be copied to the C partition of the target computer.

b. All other fields on the Options tab are used to handle special cases. For information about the Install Hook and Install Slot fields, see Chapter 9, "Specifying the module-installation order," on page 167. For information about other fields on this tab, refer to the ImageUltra help system.

18. Click the **Save** icon in the Main window tool bar; then, close the Operating System window.

19. Continue with "Building a new module" on page 166.

Setting the attributes based on an existing add-on operating-system module

To set the attributes for a new add-on operating-system module to match the attributes of an existing add-on operating-system module, do the following:

1. Open the repository where you want the new add-on operating-system module to reside.

2. If your repository contains folders, navigate to the folder where you want the new add-on operating-system to reside.

3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Operating System** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. From the list of modules, select the module you want to use as the source and then click **Next**.

10. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.

11. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.

12. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar; then, close the Operating System window.

13. Continue with "Building a new module" on page 166.

Preparing source files for a partitioning module

You use partitioning modules in a base map to prepare the drive before installing an image or to install an image itself. For example, you can use a partitioning module to erase all user partitions, erase specific user partitions, install a D partition where a user can store data files, or install a D partition that contains data. The ImageUltra Builder program provides several partitioning modules to prepare a drive (see "Using a partitioning module in a base map" on page 96 for details), but you must make your own partitioning modules to create a partition beyond the C partition. In most cases, you use an image-cloning tool, such as Symantec Norton Ghost or PowerQuest DeployCenter, to create an image of the partition, but you can also create a batch file if the partition will be empty.

The starting point for creating a partitioning module is to decide whether or not the partitioning module will contain data.

- If the partition is to contain data, you use an image-cloning tool to create an *image* of the partition. Then, you put the image of the partition, the image-cloning program, and any scripts required to install an image into a single folder.

- If the partition will not contain data, you can use an image-cloning tool to create the *scripts* required to create the partition. Then, you put the image-cloning program and any script into a single folder.

The next step is to create an operating-system module entry in the repository.

1. Open the repository where you want the new partitioning module to reside.

2. If your repository contains folders, navigate to the folder where you want the new base operating-system module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Operating System...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Operating System** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click the **No** radio button to indicate that this module is not a container.

8. Click **Next**.

9. Click **Next** without selecting any of the modules listed.

10. Click the **Partitioning module** radio button.

11. Click **Next**.

12. Click **Finish**. The New Module Wizard closes and the Operating System window for the new module opens.

13. In the General tab, do the following:

   a. In the Version field, type the full version number of the operating system.

   b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

      Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

   c. In the Comments field, type any comments that you want to keep with the module.

14. In the OS/Languages tab, do the following:

   a. In the left pane, put a check mark next to each language for which the module is intended to be used. In most cases, partitioning modules with data are language specific and only one language is typically selected. Partitioning modules that do not include data are typically language independent and multiple languages can be selected.

   b. In the right pane, put a check mark next to the operating systems for which this module applies.

15. In the Source tab, do the following:

   a. In the Source Directory field, type the complete path to the folder that contains the image files.

   b. In the Silent Install Command area, type the command required to restore the partition image or prepare the drive.

16. Click the **Save** icon in the Main window tool bar.

17. Continue with "Building a new module" on page 166.

Note: When you implement one or more partitioning modules in a base map, make sure you set the order correctly in the Install Order tab of the Map Settings window. See "Using a partitioning module in a base map" on page 96 for details.

Preparing source files for a device-driver module

Device drivers must use the standard Microsoft Plug-n-Play.INF installation method, which is usually unattended in nature. The driver itself must be certified by the Microsoft Windows Hardware Qualification Lab (WHQL) and signed. Failure to use signed, WHQL-certified device drivers will result in interruptions during the installation process by Windows-generated information messages, which require user interaction to bypass.

Adding mini-applications

The use of mini-applications is optional. Mini-applications are useful for creating special folders, cleaning up registry entries, or performing any other service related to the device-driver being installed. The content and format of a mini-application is entirely up to you. You are limited to running one mini-application before the device driver is installed and one mini-application after the device driver is installed.

The only requirement related to the preparation of mini-applications is that you must copy all of the mini-application files into the root folder of your device-driver source files.

Requirements for preparing device-driver files for modules

The following are the only requirements for preparing the device-driver source files to be made into a module:

- The device driver must be in a ready-to-install format. It cannot require an unpacking process.
- The device-driver files must be placed in a dedicated folder on a drive that is accessible by the ImageUltra Builder console.
- The device-driver folder structure must be maintained.

When the source files meet these requirements, you can set the attributes for the new device-driver module.

Setting the attributes for a new device-driver module

The following information describes how to set the attributes for a new device-driver module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing device-driver module" on page 161.

To set the attributes for a new device-driver module, do the following:

1. Open the repository where you want the new device-driver module to reside.
2. If your repository contains folders, navigate to the folder where you want the new device-driver module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Device Driver...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Device Driver** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.
7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. Click **Next** without selecting any of the modules that are listed.
10. Click **Finish**. The New Module Wizard closes and the Device Driver window for the new module opens.
11. In the General tab, do the following:
    a. In the Version field, type the full version of the device driver.
    b. In the Family field, use the drop-down menu to select the type of device for which this device driver is designed.
    c. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

       Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a

password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

    d. In the Comments field, type any comments you want kept with the module.

12. In the OS/Languages tab, do the following:

    a. In the left pane, put a check mark next to each language for which this device driver will be used. For example, you might choose to use an English device-driver module with a different language operating system module.

    b. In the right pane, put a check mark next to the operating system with which the device-driver module will be used. For example, if the device driver can be used by several Windows operating systems, check the appropriate boxes. If the device driver is specific to one operating system, check only the operating system that applies.

13. In the Source tab, do the following:

    a. In the Source Directory field, type the complete path to the folder that contains the source device-driver files.

    b. In the Silent Install Command area, leave the Command and Parameters fields blank.

> **Note:** The following steps apply to the use of mini-applications. Mini-applications are useful for creating special folders, cleaning up registry entries, or performing any other service related to the installation of the device driver. For more information about the use of mini-applications, see "Adding mini-applications" on page 158.

    c. If you have a mini-application contained in this module that needs to run immediately *before* the device driver is installed, use the Preinstall Command area to type the command and parameters associated with starting that mini-application.

    d. If you have a mini-application contained in this module that needs to run immediately *after* the device driver is installed, use the Postinstall Command area to type the command and parameters associated with starting that mini-application.

14. If you want to assign a filter to the driver module, click the Filters tab and then do the following:

    a. Click the **Insert Filter** icon. The Add Filter window opens.

    b. In the left pane of the Add Filters window, navigate to the folder that contains the filter you want to use.

    c. In the right pane of the Add Filters window, select the filter you want to use; then click **OK**. The Add Filter window closes and the filter is added to the Filters tab.

    d. Double-click the filter you just added. The Edit Parameters window opens.

    e. In the Parameters field, type any parameters required to get the desired result. For information about the parameters associated with the filters provided with the ImageUltra Builder program, see the ImageUltra Builder help system.

    f. Click **OK**.

    g. If you assign multiple filters to a module, do one of the following:

      • If *all* filters must be satisfied to install the module, click the **All of these must be true** radio button.

- If *any* of the filters can be satisfied to install the application, click the **One of these must be true** radio button.

15. In the Options tab field, do the following:
    a. The "Path to copy files on the preload" field defines whether or not you *also* want to copy the device-driver installable files to the target partition (C: or service partition). If the target partition is the C: partition, the following apply:
       - If you type a path in this field and check the **INF Installable** check box on this tab, an extra copy of the installable files will be copied to the C partition in the specified folder and the device drivers will be made available to the Windows setup or mini-setup program.
       - If you type a path in this field and do not check the **INF Installable** check box on this tab, the installable files will be copied to the C partition in the specified folder, but the device driver will not be made available to the Windows setup or mini-setup program unless you edit the default Windows Plug-n-Play path.

       Type a path only if you want the installable files to be copied to the C partition of the target computer.
    b. In the For INF install only area, check the **INF Installable** check box. If the location of the .INF file is not in the source directory you defined on the Source tab, use the "Relative location of the INF files in the source" field to type the relative path to the subdirectory that contains the.INF file. For example, if the source directory is X:\SOURCE_FILES\DRIVER_1, but your.INF file is in X:\SOURCE_FILES\DRIVER_1\WINXP, the relative path is \WINXP.
    c. All other fields on the Options tab are used to handle special cases. For information about special fields, refer to Chapter 11, "Dealing with exceptions," on page 175.
16. Click the **Save** icon in the Main window tool bar; then, close the DEvice Driver window.
17. To build a driver module, see "Building a new module" on page 166.

Setting the attributes based on an existing device-driver module

To set the attributes for a new device-driver module to the match the attributes of an existing device-driver module, do the following:
1. Open the repository where you want the new device-driver module to reside.
2. If your repository contains folders, navigate to the folder where you want the new device-driver module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Device Driver...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Device Driver** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.
7. Click the **No** radio button to indicate that this module is not a container.
8. Click **Next**.
9. From the list of modules, select the module you want to use as the source and then click **Next**.

10. Click **Finish**. The New Module Wizard closes and the Device Driver window for the new module opens.
11. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.
12. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar; then, close the Device Driver window.
13. Continue with "Building a new module" on page 166.

Preparing source files for utility modules files

The use of utilities is optional. Utilities are Win32-based EXE, BAT, or COM programs that run during the installation process. For example, an installer might want to run CHKDSK to determine the condition or size of a hard disk before any files get installed.

Utilities reside in the staging area (service partition or network-staging folder) and do not get installed on an active partition. Little preparation is required for the source files. The following are the only requirements:

- The utility must be in a ready-to-run format. It cannot require an installation process.
- The utility must be placed in a dedicated folder on a drive that is accessible by the ImageUltra Builder console.
- The utility folder structure must be maintained.

When the source files meet these requirements, you can set the attributes for the new utility.

Setting the attributes for a new utility module

The following information describes how to set the attributes for a new device-driver module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing utility module" on page 163.

1. Open the repository where you want the new utility module to reside.
2. If your repository has folders, navigate to the folder where you want the Utility module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Utility...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Utility** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.
7. Click **Next**.
8. Click **Next** without selecting any of the modules that are listed.
9. Click **Finish**. The New Module Wizard closes and the Utility window for the new module opens.
10. In the General tab, do the following:
    a. In the Version field, type the full version of the utility, if applicable.
    b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

> > **Note:** A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

> > c. In the Comments field, type any comments you want kept with the module.

11. In the Languages tab, put a check mark next to each language for which the utility is intended to be used.

12. In the Source tab, do the following:

> a. In the Source Directory field, type the complete path to the folder that contains the source utility files.

> b. In the Silent Install Command area, type the command that is to be used to start the utility in the Command field and any associated parameters in the Parameters field. For example, your command might be CHKDSK.EXE and your parameters might be /F.

13. Click the **Save** icon in the Main window tool bar; then, close the Device Driver window.

14. To build a utility module, see "Building a new module" on page 166.

Setting the attributes based on an existing utility module

To set the attributes for a new utility module to the match the attributes of an existing utility module, do the following:

1. Open the repository where you want the new utility module to reside.

2. If your repository contains folders, navigate to the folder where you want the new utility module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Utility...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Utility** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click **Next**.

8. From the list of modules, select the module you want to use as the source and then click **Next**.

9. Click **Finish**. The New Module Wizard closes and the Utility window for the new module opens.

10. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.

11. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar; then, close the Utility window.

12. Continue with "Building a new module" on page 166.

Preparing source files for filter modules files

You typically use filters to obtain hardware-related information from a target computer to determine if a specific module or set of modules should be installed. The ImageUltra Builder program provides filters that check the machine type, machine type and model, computer hardware platform (desktop or mobile), and whether or not the target computer is an IBM branded computer. You can also develop your own filters if needed.

Filters are Win32-based EXE, BAT, or COM programs that run during the installation process. In most cases, filters query the target computer BIOS for information, but filters can query any installed hardware that stores identification information in a ROM or EEPROM module. For example, you might want to check for the presence of a PCI modem before installing a modem-related application.

The filter must return a value of **1** for a true condition and a value of **0** for a false condition. A true condition results in the installation of the modules associated with that filter.

If you decide to create your own filters, you must prepare the filter files as follows:

- The filter must be able to be run from a Win32 command prompt using a single command and, optionally, a set of parameters.
- The filter must be in a ready-to-run format. It cannot require an installation or unpacking process.
- The filter must be placed in a dedicated folder on a drive that is accessible by the ImageUltra Builder console.

When the source files meet these requirements, you can set the attributes for the new filter module.

Setting the attributes for a new filter module

The following information describes how to set the attributes for a new device-driver module without using an existing module as a template. If you want to set the attributes for a new module to match the attributes of an existing module, see "Setting the attributes based on an existing filter module" on page 165.

1. Open the repository where you want the new filter module to reside.
2. If your repository contains folders, navigate to the folder where you want the new filter module to reside.
3. From the Main Window menu bar, click **Insert** and then click **Filter...**. The New Module Wizard opens.
4. Click **Next**.
5. Verify that the **Filter** radio button is selected, and then click **Next**.
6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.
7. Click **Next**.
8. Click **Next** without selecting any of the modules listed.
9. Click **Finish**. The New Module Wizard closes and the Filter window for the new module opens.
10. In the General tab, do the following:
    a. In the Version field, type the full version of the filter, if applicable.

b. If you want to assign a password to this module, type an encryption key in the "Encryption key for the password" field.

Note: A password helps prevent the module from being unpacked by any process other than an ImageUltra Builder process. When you assign an encryption key, the ImageUltra Builder program assigns a password to the module. Encryption keys can be up to 16 characters long using any combination of alpha-numeric characters. Symbols are not supported.

c. In the Comments field, type any comments you want kept with the module.

11. In the Languages tab, put a check mark next to each language for which the filter module is intended to be used.

12. In the Source tab, do the following:

a. In the Source Directory field, type the complete path to the folder that contains the source filter files.

b. In the Execution Command area, type the command that is to be used to start the filter in the Command field and any associated parameters in the Parameters field. For example, your command might be FILTERX.EXE and your parameters might be /S.

13. Click the **Save** icon in the Main window toolbar; then, close the Filter window.

14. To build a filter module, see "Building a new module" on page 166.

Setting the attributes based on an existing filter module

To set the attributes for a new filter module to the match the attributes of an existing filter module, do the following:

1. Open the repository where you want the new filter module to reside.

2. If your repository contains folders, navigate to the folder where you want the new filter module to reside.

3. From the Main Window menu bar, click **Insert** and then click **Filter...**. The New Module Wizard opens.

4. Click **Next**.

5. Verify that the **Filter** radio button is selected, and then click **Next**.

6. In the Name field, type a name that you want to use to identify this module. Be as specific as possible. This name will be the name that appears in the repository.

7. Click **Next**.

8. From the list of modules, select the module you want to use as the source and then click **Next**.

9. Click **Finish**. The New Module Wizard closes and the Filter window for the new module opens.

10. Modify any attributes that need to be changed. Refer to the ImageUltra Builder help system if you need assistance.

11. When you are finished modifying the attributes, click the **Save** icon in the Main window tool bar; then, close the Filter window.

12. Continue with "Building a new module" on page 166.

Building a new module

Before you can build a module, your source files must be prepared and accessible to the ImageUltra Builder console, and the attributes for the new module must be filled in. When these conditions are met, you can use the following procedure to build your new module:

1. Open the repository where the module entry resides.
2. If your repository contains folders, navigate to the folder that contains the module you want to build.
3. In the right pane of the Repository window, click the applicable tab (**Operating Systems**, **Applications**, **Device Drivers**, **Filters**, or **Utilities**).
4. Click the applicable module entry.
5. In the Main window menu bar, click **Tools**, and then click **Build Module...**.
6. Follow the instructions on the screen.

   Note: Some modules display a list of scripts during the build operation. Editing scripts is optional, and is typically done to troubleshoot problems or specify installation options that are not provided by the ImageUltra Builder program. For more information, see "Editing unique scripts" on page 179.

Chapter 9. Specifying the module-installation order

The ImageUltra Builder program enables you to change the order in which modules are installed. Usually, there is no need to change the default module installation order unless one or more of the following conditions exist:

- One or more modules require other modules to be installed first
- The image is not installing correctly
- You are installing a partitioning module (other than a Primary NTFS Partition partitioning module)
- Modules are interfering with each other during installation

The default installation order for modules is automatically controlled by the ImageUltra Builder program. However, you can override the default module installation order for add-on operating-system modules, Portable-Sysprep and Hardware-Specific base operating-system modules, partitioning modules, application modules, and some device-driver modules. The amount of control depends on which boot cycle the module will be installed.

Note: If you need to control the install sequence for a device-driver module, you must put the device-driver module in a base map instead of a driver map; otherwise, any settings you make for the install hook or install slot will be ignored. For more information about configuring a device driver module so it can be included in base map, see Chapter 11, "Dealing with exceptions," on page 175.

I386 base operating-system modules always install the Windows operating system first; therefore, you cannot modify the installation order of an I386 base operating-system module.

There are three methods of changing the default module-installation order:

- **Modifying the install hook:** The *install hook* defines the boot cycle (audit boot or customer first boot) in which the module is to be installed, or if the module should be installed manually through the use of a desktop icon after the image has been installed. You can modify the install hook for add-on operating-system modules, application modules, and some device-driver modules. Install hooks do not apply to Portable-Sysprep or Hardware-Specific base operating-system modules or to partitioning modules because, by design, these modules are always installed before the audit boot. Install hooks also do not apply to modules designated to be installed via a desktop icon. The install-hook setting is located in the Options tab of the module window.

  Note: The *Install hook* setting in an I386 base operating-system module is not used.

- **Modifying the install slot:** The *install slot* defines a priority grouping, 1 through 9999, where modules assigned to the slot 1 group are installed first, modules assigned to the slot 2 group are installed next, and so on until modules assigned to the slot 9999 group are assigned last. You can modify the install slot for add-on operating-system modules, application modules, and some device-driver modules. Install slots do not apply to Portable-Sysprep or Hardware-Specific base operating-system modules or to partitioning modules. The install-slot setting is located in the Options tab of the module window.

Note: The *Install slot* setting in an I386 base operating-system module is not used.

For ImageUltra Builder 3.0, the following limitations apply:

– You can use install slots 1 through 9 for add-on operating-system modules, application modules, and some device-driver modules regardless of the type of image you are installing.

– Slots 10 through 9999 are valid for use with add-on operating-system modules, application modules, and some device-driver modules *only if those modules are being installed as part of an Ultra-Portable Image based on an I386 base operating-system module.* If you attempt to assign slots 10 through 9999 to a module that is to be installed as part of a Portable-SysPrep Image, Hardware-Specific Image, or an Ultra-Portable Image based on an imported ImageUltra-developed base operating-system module, the slot setting will be ignored.

• **Modifying the install order within a base map:** The Install Sequence window enables you to modify the following:

– The order in which modules that share an install slot are installed during the first customer boot

– The order in which base operating-system modules and partitioning modules are installed relative to other base operating-system modules and partitioning modules contained in the map.

– The order in which modules are installed in the service partition.

The installation-order process

The process for changing the module-installation order depends on the type of modules you need to control:

• If you are working with an I386 base operating-system module, the installation-order process does not apply because I386 base operating-system modules do not go through a formal installation process. However, the installation order for other modules contained in the same map as an I386 base operating-system module can be changed.

• If you are working with add-on operating-system modules, application modules, or device driver modules, see "Add-on operating-system modules, application modules, and device-driver modules."

• If you are working in a map that contains both base operating-system modules and partitioning modules, see "Partitioning modules" on page 169.

Add-on operating-system modules, application modules, and device-driver modules

If you are working with add-on operating-system modules, application modules, or device-driver modules that get installed from a base map, there are three primary steps to determine the entire order of installation:

1. Modify the install hook to establish the boot cycle. (See "Modifying the install hook" on page 169 for more information.)

2. Modify the install slot to establish the installation order within the boot cycle defined by the install hook. (See "Modifying the install slot" on page 170 for more information.)

3. Modify the install sequence in the Map Settings window to fine tune the installation order within an install slot for modules that are installed during the customer-first-boot cycle. (See "Modifying the install sequence with a base map" on page 170 for more information.)

It is important that you understand that the default settings are different for imported ImageUltra modules and modules created through the ImageUltra Builder program.

- By default, all *imported ImageUltra-developed* application modules and add-on operating-system modules are installed during the *audit boot*. The audit boot is the first boot cycle after Windows setup is complete and occurs before the customer first boot.
- By default, all application modules and add-on operating-system modules *created through the ImageUltra Builder program* are installed during the the *customer first boot*.

The ImageUltra Builder program enables you to control the installation order for modules that are installed during the audit boot and modules that are installed during the customer first boot. One benefit of keeping modules in the audit boot is that it typically decreases the time needed during the customer first boot. One benefit of defining modules to be installed during the customer first boot is that you can control the installation order within an install slot, a benefit you do not have with modules installed during the audit boot.

Partitioning modules

If you are working with a map that contains one or more partitioning modules, you must do the following:

- If the only partitioning modules used in a map are Primary NTFS Partition partitioning modules, no modifications to the install sequence are required.
- If you are using any other type of partitioning module you must modify the install sequence in the Map Settings window to ensure the following:
  - The partitioning modules are installed before any base operating-system module that yields an Ultra-Portable Image; otherwise, the partitioning module might not be installed.
  - The partitioning modules are installed in the correct order to get the desired result.

See "Modifying the install sequence with a base map" on page 170 for more information.

Modifying the install hook

If you are working with an add-on operating-system module, application module, or a device-driver module that gets installed from a base map, you can modify the install hook. Modify the install hook in the Options tab of the module window. The following is a list of choices available from the Install Hook drop-down menu:

- **Install during audit boot (Default):** This selection enables the module you are creating to be installed during the first boot *after* the Windows setup is complete.
- **Install during customer first boot:** This selection enables the module you are creating to be installed after the audit boot, upon the successful completion of Windows setup.
- **Install when customer chooses via desktop shortcut:** This selection enables users to install the module anytime after the image is installed by clicking a shortcut icon on the desktop. In this case, the ImageUltra Builder program copies the module source files to a folder on the user partition and creates a shortcut icon on the desktop that links to the installation file.

To modify the install hook, do the following:

1. From the Repository window, double-click the module entry. The module window opens.
2. Click the **Options** tab.
3. Use the **Install hook** drop-down menu to select the appropriate install hook.
4. In the Main window toolbar, click the **Save** icon.
5. Close the module window.

Modifying the install slot

If you are working with an add-on operating-system module, application module, or device-driver module that gets installed through a base map, you can modify the install slot.

The Image Ultra Builder program automatically assigns install slots to control the order of installation. The install-slot number determines when a module is installed in relation to other modules that are assigned during the same boot cycle (as defined by the *install hook*). Install slot numbers range from 1 to 9999. The higher the slot number, the later the module is installed during the boot cycle. Install slots 1 through 5 are most commonly assigned by the ImageUltra Builder program. So, if you need to ensure that a module is installed late in the boot cycle, assign an install slot higher than 5. If you have a dependency on one module being installed before another module, assign a lower install-slot number to the module that needs to be installed first. Modify the install slot in the Options tab of the module window.

To modify an install slot, do the following:
1. From the Repository window, double-click the module entry. The module window opens.
2. Click the **Options** tab.
3. Use the **Install slot** drop-down menu to select the appropriate install slot number.
4. In the Main window toolbar, click the **Save** icon.
5. Close the module window.

Modifying the install sequence with a base map

You can modify the install sequence in a base map to accomplish either of the following:
- To fine tune the installation order for modules that get installed *during the customer first boot and share the same slot number*
- To set the installation order for one or more partitioning modules and a base operating-system module when both are being used in the same map.

In the Install Sequence window, use a drop-down menu to select a customer-first-boot module (by install slot) or base operating-system modules; then, set the install sequence for all modules within that type.

The Set the Install Sequence fields are as follows:
- **First:** Place modules that need to be installed first in this field in a specified sequence.
- **No Preference:** Place modules that do not require sequencing in this field in any order.

- **Last:** Place modules that need to be installed last in this field in a specified sequence.

To modify the installation sequence within a base map, do the following:

1. In the Repository window, double-click the appropriate base-map entry. The Base Map window opens.
2. In the Main window menu bar, click **Tools**, and then click **Install Sequence**. The **Install Sequence** window opens.
3. Select the type of modules for which you want to set the order. The modules for the selected type are shown in the **Set the install sequence** boxes.
4. Click the module entry you want to move, and then use the **Move Item Up** or **Move Item Down** icon in the Map Settings window to move the module entry to the appropriate box (**First**, **Last**, or **No preference**). Repeat this step for each module you need to move.
5. After all modules have been moved to the appropriate box, use the **Move Item Up** or **Move Item Down** icon in the Map Settings window to set the order within the **First** box and the **Last** box.

   Note: When working with partitioning modules, adhere to the following:
   - By default, Primary NTFS Partition partitioning modules (*Win2000 Primary Partition*, *WinXP Home Primary Partition*, or *WinXP Pro Primary Partition*) are always installed before their associated I386 base operating-system module. In most cases, there is no reason to move Primary NTFS Partition partitioning modules.
   - When working with I386 base operating-system modules, partitioning modules designed to create or preserve extra partitions (such as *Drive Prep - Delete C Partition only*) must be installed before Primary NTFS Partition partitioning modules (*Win2000 Primary Partition*, *WinXP Home Primary Partition*, or *WinXP Pro Primary Partition*). Otherwise, the partitioning module might not be installed.
   - Partitioning modules designed to create or preserve extra partitions (such as *Drive Prep - Delete C Partition only*) must be installed before any imported ImageUltra-developed base operating-system module that yields an Ultra-Portable Image; otherwise, the partitioning module might not be installed.
   - When working with multiple partitioning modules, verify that partitioning modules are installed in the correct order to get the desired result.

6. Repeat steps 3 through 5 for each module type you need to modify.
7. Click **OK**.
8. Click the **Save** icon in the Main window toolbar.

Installation-order tips and notes

Tip: If you are installing an Ultra-Portable Image or a Portable-Sysprep Image that contains the Windows XP operating system, a log is generated that identifies the types of modules that were installed and identifies when the modules were installed. This log is helpful in determining if your image has modules that are dependent on one another, or if modules are colliding. The Modules.log file is located in the C:\IBMTOOLS folder. In addition, another log file, called stamp log, is located in the C:\windows\lastcfg directory.

Notes:

- As modules are installing, the environment state is saved so that if a controlled reboot occurs, module installation will continue where it left off.
- When a new first-customer-boot module is added to a base map, it is automatically placed in the "No Preference" category of the installation sequence for its designated install slot.
- If a base map is constructed such that a partitioning module will be installed with an imported ImageUltra-developed base operating-system module (for an Ultra-Portable Image), the Map Settings window must be configured so the partitioning module is installed before the base operating-system module. There is no exception to this rule.
- If a base map is constructed such that:
    - A partitioning module will be installed with a base operating-system module developed for either a Hardware-Specific Image or a Portable-Sysprep Image
      -AND-
    - The partitioning module will never be installed with a base operating-system module developed for an Ultra-Portable Image

  you can technically set the installation sequence in the Map Settings window any way you want. However, there are advantages to configuring the Map Settings window so the partitioning module is installed before the base operating-system module. In most cases, you define the extended partition defined in the partitioning module to occupy a fixed portion of the hard disk or a fixed percentage of the hard disk. This enables you to define the primary partition contained in the base operating-system module to occupy whatever space is left on the hard disk. By following this practice, you can install the same base operating-system module with or without a base partition module and use the maximum amount of space that is available on the hard disk.

Chapter 10. Getting help and support

The ImageUltra Builder program is supported by Lenovo. If you have a problem with the ImageUltra Builder or have questions about a specific feature, a variety of sources are available to help you including documentation, an integrated help system, and Web pages.

Support for third-party tools that might be used in conjunction with the ImageUltra Builder (such as Symantec Norton Ghost, PowerQuest DeployCenter, or the Microsoft Sysprep program) is provided by the manufacturers of those tools, not Lenovo.

Refer to your ImageUltra Builder license for a full explanation of Lenovo support, terms, and duration. To access the license, do the following:

1. Start the ImageUltra Builder program.
2. From the Main window, click **Help**, and then click **View the license**.

Be sure to retain your proof of purchase.

Note: ImageUltra Builder is not a warranted product. Lenovo will provide assistance to resolve problems and answer questions about the ImageUltra Builder, but Lenovo is not obligated to update this version of the code to enhance its functionality or resolve problems, or make future versions of this product available at no charge. Any decision to make updates available or to charge fees for updates is solely at the discretion of Lenovo.

Using the documentation and help system

Many problems can be solved without contacting Lenovo for assistance. If you experience a problem or have a question about the operation or functionality of the ImageUltra Builder program, begin with the online help system or this publication, the *ImageUltra Builder User Guide*.

For example, if you need assistance with conceptual information, use the *ImageUltra Builder User Guide*. If you need help with a specific task or procedure, use the help system. To access the help system, do one of the following:

- Press **F1**
- Click **Help** in the Main window menu bar, and then click **View help topics**
- Click the Help icon in the Main window toolbar

The help system will open to a topic related to the currently-active ImageUltra Builder window. You can select other topics from the contents, or use the search or index functions to find a particular word or phrase.

Using the Web

The ImageUltra Web site provides the latest technical information, any downloadable updates that Lenovo chooses to distribute, and a listing of HIIT-enabled IBM branded personal computers. To visit the ImageUltra Web site, go to `http://www.lenovo.com/think/support/site.wss/MIGR-44316.html`.

Contacting an ImageUltra Builder technical expert

Technical assistance for the ImageUltra Builder is available through the ImageUltra Web site. You can contact the ImageUltra technical expert through e-mail service provided on the ImageUltra Web site. To access the Web site, go to

`http://www.lenovo.com/think/support/site.wss/MIGR-44316.html`

, then click **Submit a request**.

Chapter 11. Dealing with exceptions

The ImageUltra Builder program is designed to be extremely flexible. This design encourages exploration as you and your team become comfortable building images and modules. As your image-building process evolves, there might be times when you want to accomplish tasks not covered in the previous chapters or in the ImageUltra Builder help system. The purpose of this chapter is to provide you with solutions to special exceptions that you might encounter as you develop your image-building process.

The information in this chapter provides an overview on how to deal with exceptions. For detailed step-by-step instructions about working with map settings or module attributes, refer to the ImageUltra help system.

Application module exceptions

There are several ways that application modules are created to store important information that is not directly related to application source files. The following conditions explain some of the unique exceptions pertaining to applications.

Condition: Adding non-installable application files (or files of any kind) to Ultra-Portable and Portable-Sysprep Images.

Solution: The ImageUltra Builder program enables you to build an application module that simply copies files to a specific location on the target computer hard disk. This method is optional and cannot be used for Hardware-Specific Images.

The process for creating this type of application module is similar to creating a standard application module. However, there are slight differences when you specify the attributes. After you have your source files prepared, use the New Module Wizard to create a new application module as you would a standard application module, but allow for the following exceptions when you fill in the attributes:

- In the **Source** tab, fill in the **Source Directory** field with the complete path to the folder containing the non-installable files for the module you are creating
- In the **Source** tab, do not enter any commands for the **Silent Install Command** field, the **Preinstall Command** field, or the **Postinstall Command field**.
- In the **Options** tab, type the full path to where the files are to be copied in the **Path to copy installable files** field.
- In the **Options** tab, use the **Target Partition** drop-down menu to select the appropriate partition where the files are to be copied.

After specifying attributes for the application module, click the **Save** icon located in the Main window toolbar. Then, build the module.

Condition: Creating batch files to install applications for Ultra-Portable and Portable-Sysprep Images.

Solution: Certain applications require significant logic in order to install correctly. The ImageUltra Builder program enables you to install these types of applications using a batch file. This method is optional and cannot be used for Hardware-Specific Images.

The process for creating an application module that uses a batch file for installation is similar to creating a standard application module. However, there are slight differences when you specify the attributes. After you have your source files prepared, use the New Module Wizard to create a new application module as you would a standard application module, but allow for the following exceptions when you fill in the attributes:

When creating an application module that contains a batch file, create the module as if it were a standard application module, taking into consideration the following exceptions:

- In the **Source** tab, complete the **Source Directory** field with the entire path to the folder containing the batch file and related application files for the module you are creating
- In the **Source** tab, complete the **Silent Install Command** field with the name of the batch file
- In the **Options** tab, type the full path to where the files are to be copied in the **Path to copy installable files** field.

Note: Using a batch file to install an application requires that the application files and the batch file are the only files that exist in the directory. You can have as many batch files as needed, but only one batch file will be executed from the silent install command.

After specifying attributes for the application module, click the **Save** icon located in the Main window toolbar. Then, build the module

Condition: Providing hardware detection for applications that do not have detection capabilities. For example, you might have an application that is needed only if a specific hardware device is present, but the application does not have hardware detection capabilities. An example of this type of application is a CD-RW application that does not have hardware detection.

Depending on the level of hardware detection required, there are two possible solutions.

Solution 1: If the application is unique to a specific vendor PCI device, you can assign the *ImageUltra 3.0 PCI - Adapter Check* filter to your application module. This filter is provided as part of the ImageUltra Builder program and checks for the presence of a specific 12-digit PCI vendor device ID. The filter parameter format for this filter is *VVVVVV DDDDDD*, where *VVVVVV* is the vendor ID and *DDDDDD* is the device ID. For more information about associating a filter with an application module, see "Setting the attributes for a new application module" on page 151 or the ImageUltra Builder help system.

Solution 2: If you already have a separate program to detect hardware, or if you have the resources to develop one, the ImageUltra Builder program enables you create an application module that includes a Win32 program that does hardware detection and then installs the application, if it is needed.

The process for creating this type of application module is similar to creating a standard application module. However, there are slight differences when you specify the attributes. After you have your source files prepared, use the New Module Wizard to create a new application module as you would a standard application module, but allow for the following exceptions when you fill in the attributes:

- In the **Source** tab, complete the **Source Directory** field with the entire path to the folder containing the hardware detection program and related application files for the module you are creating.
- In the **Source** tab, complete the **Silent Install Command** field with the name of your hardware detection program that first searches for the hardware and then installs the application if the hardware is present. Do not enter commands for the **Preinstall Command** field, or the **Postinstall Command** field.
- In the **Options** tab, type the full path to where the files are to be copied in the **Path to copy installable files** field.

After specifying attributes for the application module, click the **Save** icon located in the Main window toolbar. Then, build the module.

Condition: Problems installing modules for applications with very deep folder structures

Solution: Do the following:
1. Create a self extracting .ZIP file from your source files.
2. Create a setup batch file that does the following:
   - Unzips the .ZIP file
   - Runs setup using a script file
3. Put all files in a common folder; then, create the module.

Device driver exceptions

You can manipulate device-driver modules for various types of device drivers *and* applications. The following conditions explain some of the exceptions pertaining to device drivers.

Condition: The installation order of a device-driver module needs to be modified to resolve a problem.

Solution: Under certain conditions, you can install a device-driver module through a base map to control the installation order. No modifications are required to the source files or module.

Condition: Pointing to an .INF file that exists in a source file subdirectory.

Solution: Typically, when building a device-driver module, the .INF file is located in the root directory with other source files used for that device driver. In some cases, a device driver might contain an .INF file in a subdirectory. The ImageUltra Builder program allows for .INF files in subdirectories.

The process for creating a device-driver module that points to an .INF file subdirectory is similar to creating a standard device-driver module. However, there is one slight difference when you specify attributes. After you have your source files prepared, use the New Module Wizard to create a new driver module as you would a standard device-driver module, but allow for the following exception when you fill in the attributes:

- In the **Options** tab, complete the **Relative location of INF files in the source** field with the relative path to the subdirectory that contains the .INF file. For example, if the source file is X:\SOURCE_FILES\DRIVER_1, but your .INF file is in X:\SOURCE_FILES\DRIVER_1\WINXP, the relative path is \WINXP.

After specifying attributes for the device-driver module, click the **Save** icon located in the Main window toolbar. Then, build the module.

Condition: Preparing a device-driver module that installs using an .INF file, but also has an application executable packaged with it.

Solution: Some applications are hardware dependent. These applications are sometimes designed to contain *both* device-driver *and* application files. For example, you might have source files that contain a wireless device driver intended for a wireless network card and a configuration utility intended to work in conjunction with the network card.

In this case, if you make an *INF-only device-driver module* from the application source files, the application might fail to install correctly. Conversely, if you make an *application module* from the application source files, the device driver might fail to install correctly.

Although there are several ways to address this exception, the easiest and most successful solution is to make a device-driver module that is *both* an INF installable *and* an executable. This method is optional and cannot be used for Hardware-Specific Images.

The process for creating this type of device-driver module is similar to creating a standard device-driver module. However, there are slight differences when you specify the attributes. After you have your source files prepared, use the New Module Wizard to create a new device-driver module as you would a standard device-driver module, but allow for the following exceptions when you fill in the attributes:

- In the **Options tab**, check the **INF installable** check box.
- In the **Source tab**, complete the **Silent Install Command** field with the command used to install the application.

After specifying attributes for the device-driver module, click the **Save** icon located in the Main window toolbar. Then, build the module. After building the module, insert the module into the appropriate driver map. When the module is installed with the image, the .INF files are copied first into the Windows driver directory, and then the silent install command is executed during the Windows audit boot.

Making allowances for modules that require a reboot

Occasionally, you might encounter a condition where an application, add-on operating-system component, or device driver must trigger a reboot cycle to complete its installation. The ImageUltra Builder program provides a **Reboot after install** check box on the Options tab of the module window. When the **Reboot after install** check box is marked, the ImageUltra Builder program does the following during the installation phase:

1. Records the state of the overall installation process
2. Initiates a reboot cycle at the end of the module installation

After the reboot cycle, the overall installation process continues where it left off.

To instruct a module to reboot after install, do the following:

1. Open the appropriate repository.

2. If your repository contains folders, navigate to the folder that contains the appropriate module.

3. In the right pane of the Repository window, select the tab for the appropriate module type.

4. Double-click the appropriate module. The module window opens.

5. Click the **Options** tab.

6. Mark the **Reboot after install** check box.

7. Click the **Save** icon in the Main window toolbar.

8. Close the module window.

Editing unique scripts

The ImageUltra Builder program enables you to specify whether or not you want to view, and edit, build scripts whenever you build a module. In most cases, the program generates the build scripts by processing information that is contained in the module attribute tabs. Editing scripts is optional, and is typically done to troubleshoot problems or specify installation options that are not provided by the ImageUltra Builder program.

Important: Base operating-system modules for Portable-Sysprep Images and Hardware-Specific Images do not generate a build script when you build a module; nor does a partitioning module. All other types of modules do generate a build script during the build process.

Condition: Editing Unique Scripts

Solution: If you choose to edit build scripts, some of the file extensions might be familiar while others might not. The following is a list of unique files that contain build scripts and a brief description of each:

- LPL,WPL, and BAT files: These files are run during WinPE mode (immediately before the Windows installation begins, even before the audit boot). Editing these scripts requires using Win32 commands only.

- CMD files: These files are executed during the Windows audit boot. Editing these scripts requires using Windows commands only. CMD files apply to Windows XP installations only, and only if Windows XP is checked in the OS/Language tab for that particular module.

- FM and FM2 files: Generally, these files are used by the Fmodify.exe program. This program is used to automatically modify the contents of other files. In most cases, an FM2 file functions similarly in Windows 2000 as a CMD file functions in Windows XP.

  Notes:

  1. If you are not familiar with the Fmodify.exe program, it is available in the tools directory of the module repository and is made when the repository is created. To get more information regarding editing FM file scripts, go to the tools directory and enter the following command: `modify /?`

  2. If you rebuild a module after it is edited, then you have to edit the scripts again the next time you build the module. It is a good idea to track your edits carefully and record them in a separate file for safekeeping.

Including the ThinkVantage Rescue and Recovery product

The ThinkVantage Rescue and Recovery product is available three ways:

- A Web downloadable version licensed for ThinkPad, ThinkCentre, and NetVista computers only. This version does not use a physical service partition. All critical recovery files are stored in a virtual service partition, which resides on the C drive. The virtual service partition is hidden and behaves just like a physical service partition, however installation is simpler and faster than possible with Rapid Restore PC because data does not have to be restructured to create a new service partition or resize an existing service partition. This version contains a BIOS check to verify that the computer it is being installed on is in fact an Lenovo branded computer. You can download the installable files from: http://www.lenovo.com/think/support/site.wss/MIGR-4Q2QAK.html.
- A retail version with per-seat licenses for non-IBM branded computers: This version is functionally identical to the Web-downloadable version except it does not contain the BIOS check. Therefore, it can be installed on practically any personal computer. For information about purchasing the retail version, contact your Lenovo Marketing representative.
- A preinstalled version on many new ThinkPad and ThinkCentre computers: This version uses a physical service partition for many of the recovery and diagnostic tools as well as Windows-based files for the Rapid Restore backup tool and recovery CD creation tools. Do not attempt to import and use Rescue and Recovery modules from the service partition of a new ThinkPad or ThinkCentre computers. Instead, use the Web downloadable version or the retail version.

This version of the ImageUltra Builder program does not support importing the modules associated with the ThinkVantage Rescue and Recovery product that comes preinstalled on select IBM-branded personal computers. You must create your own modules using either the Web-downloadable version (for use with IBM-branded computers only) or the retail version (for use with IBM branded computers and other computer brands) of the ThinkVantage Rescue and Recovery product. The Web-downloadable version is available from the Lenovo Web site at http://www.lenovo.com/think/support/site.wss/MIGR-4Q2QAK.html.

Notes:

1. If you intend to create an image that includes either the Web-downloadable version or retail version of the ThinkVantage Rescue and Recovery product, you must define a custom service partition size that leaves a fixed buffer of at least 500 MB through the Partitions tab of the Base Map settings.

2. By default, the Web-downloadable version and retail version of the ThinkVantage Rescue and Recovery product does not create a physical service partition as part of its installation process; instead it creates a virtual service partition on the C drive.

   However, when installed under the control of the ImageUltra Builder program, these versions of the ThinkVantage Rescue and Recovery product detect the presence of the physical service partition created by the ImageUltra Builder program, and install themselves in the physical service partition. If you install the ThinkVantage Rescue and Recovery product as part of a Smart Image, do not use the Delete all setting in the Image Cleanup tab of your base map.

3. The ThinkVantage Rescue and Recovery product can not be preinstalled on the donor system before creating a portable Sysprep image. It must be installed as a separate application module.

Controlling write-access to your repository during a direct-network install

By default, the target computer has read access and write access to your repository during a direct-network install operation. The write access is required so the target computer can make modifications in the network-staging folder, which by default is located in the repository. For those of you who do not want the target computer to have write access to your repository, you can specify a location for the network-staging folder outside of the repository. Taking this precaution gives the target computer read-only access to the repository and read/write access to the network-staging folder that resides outside of the repository.

To specify a location outside of the repository for the network-staging folder, do the following:

1. Use Windows Explorer or My Computer to create a folder on your network that will be used as a network staging folder.
2. From the ImageUltra Builder Main window, open any repository.
3. From the Main window menu bar, click **Tools** and then click **Options...**. The Repository Options window opens.
4. Click the **Network Install** tab.
5. In the Location field, either use the Browse button to locate the folder you created in Step 1, or type the location of the folder in Universal Naming Convention (UNC) format.
6. In the Server IP Address field, type the IP address of the server that contains the folder specified in the Location field. (Optional)
7. Click **OK**.

All network-staging folders will be created dynamically under the specified folder.

Chapter 12. Debugging problems

Occasionally, you might have to debug problems during the test phase of your development process. This chapter provides information about some of the debugging tools provided by the ImageUltra Builder program and some best practices to follow to avoid some problems that are difficult to debug.

The ImageUltra Builder image-installation process

To debug a problem, you must have some level of understanding of the processes that take place during the image-installation phase. All files referenced in this section are located either in the service partition of the target computer, or, in the case of a direct-network installation, in the network-staging folder.

The AUTOEXEC.CMD file is the first program launched at the beginning of the image-installation phase and acts as the master control program for all activities during the image-installation phase. Activities include the following:

1. Establishing global variables and PATHs.
2. Controlling the installation menus. The Mapmenu program runs and processes the Before Menu Options. Once the Before Menu items are complete, all of the menu items in the basemap are displayed. Once a selection is made from the basemap, the Post Menu options are processed, and the EXECUTE.CMD file is written into the IUTEMP directory.
3. Once the menu processing is complete, the EXECUTE.CMD file runs to process and queued utilities from the basemap.
4. All of the selected modules are written into the AOD.DAT and the process installs all of the modules in this file.
5. Customer-first-boot module processing. Upon the successful completion of the audit boot, the modules in the \IUWORK folder are processed.

Finding debug data in the target computer user partition

During the image-installation process, the ImageUltra Builder program creates a MODULES.LOG file in the \IBMTOOLS folder of the target computer user partition. The MODULES.LOG provides a listing of each module that installed successfully. If a problem occurs during installation, you can use this log to detect the last module that installed successfully, and use that as a starting point for debugging.

Finding debug data in the service partition or network-staging folder

During the image-installation process, the ImageUltra program creates several logs in the IUTEMP folder of staging area (service partition or network-staging folder). If an error occurs during the installation process, the installation process is halted, an error message is posted, and you can exit from the installation process and view the logs to determine where the error occurred.

The following is a listing of those logs:
- The IUTEMP\STAMP.LOG: log contains time-stamped entries for all activities performed in the staging area

To get to a command prompt, press and hold the **Ctrl+Alt+Shift** keys, and then click **Restart**. To view the STAMP.LOG file, type the following at the command prompt:

```
notepad %SystemDrive%/Recovery/IUB/IUTEMP/STAMP.LOG
```

If you are performing a standard network-deployment or deployment from CD or DVD, the debug data is located in the service partition of the target computer. The volume label of the service partition is IBM_SERVICE. The service partition is not hidden until the image-installation process has completed successfully. Therefore, if you encounter an error during the installation, the service partition is accessible.

If you are performing a direct-network installation, the debug data is located in the network-staging folder. By default, the network-staging folder is located under the repository being used for the installation and is named \NETDEPL*xxxxxxxxxxx*, where *xxxxxxxxxxx* is the last 11 characters of the target computer MAC address. You can, however, change the location of the network-staging folders to be outside of your repository (see "Controlling write-access to your repository during a direct-network install" on page 181 for details). The network-staging folder is not erased until the image-installation process has completed successfully. Therefore, if you encounter an error during installation, you can access the network-staging folder from any computer that has access to the repository or to the location where an external network-staging folder resides.

Monitoring the installation

During the image-installation phase, a "splash" screen is displayed, showing a progress bar. You can disable this splash screen and monitor the installation by pressing **Shift+D** anytime during the image-installation process. When you disable the splash screen, you can monitor the execution of the current command.

Splash Utility

This utility displays a full screen, always on top window called a *splash* window. This window appears in front of all other application windows, with the exception of the window created by MAPMENU.EXE, the ImageUltra map menu viewer. In addition, while the splash window is displayed, the following standard Windows task-switching keystroke combinations are disabled:

- CTRL+TAB.... Switch to another running application via task window
- CTRL+ESC.... Opens the start menu.
- ALT+ESC.... Switches to the next application.
- WINDOWS keys.... Miscellaneous shell navigation functions.

SPLASH.EXE also turns off the Windows taskbar, which is always on the top of the display (only while the splash window is displayed). This prevents the taskbar from appearing in front of the splash window as other application windows are potentially activated. The splash window is used to display the contents of a log file. A message-box dialog displays messages to users as it moves around the screen.

The syntax of the SPLASH.EXE program is as follows:

SPLASH.EXE [/STart] [/SHutdown] [/MINimize] [/MAXimize]
/Settings=<path>/Text=<path>

Parameters	Description
/STart	Displays the splash window maximized. If a previous instance of SPLASH.EXE is already running, the program simply exits.
/SHutdown	Closes an existing splash window, if one exists, then exits.
/MINimize	Displays the splash window minimized. If a previous instance of SPLASH.EXE is already running, the program minimizes the existing splash window and then exits.
/MAXimize	Displays the splash window maximized. If a previous instance of SPLASH.EXE is already running, the program maximizes the existing splash window and then exits.
/Settings	Specifies path to settings file, which controls appearance of splash screen and animated message box (see notes below). Default: SPLASH.INI
/Text	Specifies path to text strings file, which contains the title and text of the animated message box (see notes below). Default: none.

If no command switch (i.e. /STart, /SHutdown, /MINimize or /MAXimize) is specified, the default behavior is the same as if /MAXimize had been specified.

The settings file controls numerous aspects of the splash window and message box dialog appearance and behavior. This file is in the format of a Windows INI file, as shown below:

Settings	Values
WinDebugKey	A keystroke sequence which will alternately minimize and maximize the splash window. This can be used to assist in debugging the deploy process. In addition to the alphanumeric keys (A-Z and 0-9), the following keys are recognized: • CTRL • ALT • SHIFT • BACK • TAB • CENTER • PAUSE • ESCAPE • SPACE • PAGEUP • PAGEDOWN • END • HOME • LEFT • UP • RIGHT • DOWN • PRTSCREEN • INSERT • DELETE • PLUS • COMMA • MINUS • PERIOD

	- OEM1
	- OEM2
	- OEM3
	- OEM4
	- OEM5
	- OEM6
	- OEM7
	- F1
	- F2
	- F3
	- F4
	- NUMPAD6
	- NUMPAD7
	- NUMPAD8
	- NUMPAD9
	- MULTIPLY
	- ADD
	- SEPARATOR
	- SUBTRACT
	- DECIMAL
	- DIVIDE
	- NUMLOCK
	- SCRLOCK
	
	Keys must be concatenated with the + character, for example:
	`CTRL+ALT+SHIFT+F12`
	    `Default: none`

`Stopfile`	Specifies the path to a file that will cause the application to terminate once the file exists. `Default: none`
`LogFile`	Specifies the path to a log file, the contents of which are displayed within the splash window. The application continually monitors this file for changes and updates the splash window. `Default: none`
`FontName`	The name of the font to be used to display text within the splash window or message box (e.g. COURIER). Default: Lucida Console (splash window), MS Sans Serif (message box)
`FontPointSize`	The point size of the font to be used to display text within the splash window or message box (e.g., 12) Default: 10 (splash window), 8(message box)
`ForeGroundColor`	The color of the text within the splash window or message box. Default: GREEN (splash window), WINDOWTEXT (message box)
`BkgndColor`	The background color of the splash window or message box[1]. Default: BLACK (splash window), BUTTONFACE (message box)

`Width`	Specifies the width of the message box dialog. If values are not specified for both Width and Height, the message box window is auto-sized based on the message text. `Default: none`
`Height`	Specifies the height of the message box window. If values are not specified for both Width and Height, the message box window is auto-sized based on the message text. `Default: none`
`Timer`	Specifies the pace, in seconds, at which the position of the message box dialog is changed. `Default: 1`
`BkgndImage`	Specifies the path of a Windows bitmap (bmp) file to be displayed within the background of the message box dialog. `Default: none`
`BkgndImagePos`	Specifies the location of the message box dialog's background image. The value of this setting is one horizontal position value (LEFT, RIGHT, or CENTER) and one vertical position value (TOP, CENTER, or BOTTOM) separated by a space (e.g., RIGHT BOTTOM). Default: RIGHT BOTTOM

[1]Color values may be specified as named values supported by MS Internet Explorer (e.g., *RED, FUSCHIA, PINK, etc.); as RGB triplets (e.g., FF0000); or as the user-defined system settings i.e., ACTIVEBORDER, ACTIVECAPTION, APPWORKSPACE, BACKGROUND, BUTTONFACE, BUTTONHIGHLIGHT, BUTTONSHADOW,BUTTONTEXT, CAPTIONTEXT, GRAYTEXT, HIGHLIGHT, HIGHLIGHTTEXT, INACTIVEBORDER, INACTIVECAPTION, INACTIVECAPTIONTEXT, INFOBACKGROUND, INFOTEXT, MENU, MENUTEXT, SCROLLBAR, THREEDDARKSHADOW, THREEDFACE, THREEDHIGHLIGHT, THREEDLIGHTSHADOW, THREEDSHADOW, WINDOW, WINDOWFRAME, WINDOWTEXT).*
 	
The text file identifies the message box dialog's title and message. (These values are separated from the settings file to facilitate national language support). This file is in the format of a Windows INI file, as shown below:
 	
`[MsgBox]`
`Title=`
`Text=`	

Title	Message box dialog's caption text
Text	Message box dialog's message text

Sending debug data to the technical support team

If you need assistance in debugging a problem, zip the entire \IUTEMP folder in the staging area and the \IUWORK folder in the user partition. You can use the PKZIP25 or your own ZIP program for the following task:

Drive letter limitations and the use of variables

The ImageUltra Builder program has a number of predefined global variables that you can use in batch files or in the Command and Parameter fields in the ImageUltra Builder interface. It is a good idea to avoid hardcoding drive letters and to use the provided global variables instead because the drive letter designations can change based on whether or not you perform a direct-network installation.

If you are performing a standard installation (network or CD/DVD deployment and local installation), drive-letter variables are converted to single-digit drive letters. However, if you are performing a direct-network installation, where a network-staging folder is used, some drive-letter variables are converted to a path pointing to the network staging folder, which could be very long.

The following are predefined global variables for the staging area:
- %ROOT%: This variable is the root path for all ImageUltra Builder deployment and installation code. The default value is C:
- %ROOTDRIVE%: This variable is the drive letter of the root path. The default value is C:
- %IUTEMP%: This variable defines the temporary work area. The default value is C:\IUTEMP
- %IUTEMPDRIVE%: This variable is the drive letter of the path for the temporary work area. The default value is C:
- %NETINSTALL%: If the NETINSTALL variable is defined in any way, the process assumes that it is running from the network.
- %RAMD%: This variable defines the RAM drive used by the ImageUltra Builder installation process. The default value is R:
- %SPDRIVE%: This variable defines the drive letter of the service partition. The default value is C:
- %IULOG%: This variable defines the folder where the SNAPSHOT.BAT file moves the %IUTEMP% at the end of the ImageUltra Builder installation process. The default value is C:\LASTCFG.

The following are predefined variables for the \IUWORK folder on the user partition:
- %SRCLOC%: This variable defines the location of the IUWORK folder and is available only when the ImageUltra Builder program is performing installation procedures under Windows control. The default value is C:\IUWORK.
- %OSTYPE%: This variable is the operating system that is currently installed, and is available only when the ImageUltra Builder program is performing installation procedures under Windows control.
- %TIMESTAMPLOG%: This variable defines the event log file created by the ImageUltra Builder installation processes. All major events are time stamped in this file. The default value is STAMP.LOG

Use these predefined global variables for their specified purpose only.

F11 key does not work on some brands after installation

After an ImageUltra Builder deployment and installation, pressing F11 during startup typically provides access to the ImageUltra menu system, either directly or indirectly. However, you might encounter a brand or line of computers where pressing F11 during startup has no effect after an ImageUltra Builder deployment and installation. Such cases are rare, but they do exist.

You can use either or both of the following methods to work around this problem:

Method 1: Add an shortcut icon on the Windows desktop using the following procedure.

1. Use Windows Explorer or My Computer to locate the BMGR32.EXE file located in the \TOOLS\BMGR folder of any repository.
2. Copy the BMGR32.EXE file to user partition of the target computer that is failing. You can place this file anywhere on the user partition.
3. Create shortcut icon on the desktop that contains the following command:
   ```
   path\bmgr32 /BS /R
   ```
   where *path* is the path to the folder containing the BMGR32.EXE file.

The user can use the shortcut icon to access the ImageUltra menu system.

Method 2: Create a recovery CD using the following procedure.

1. Create a bootable DOS CD.
2. Use Windows Explorer or My Computer to locate the BMGR.EXE file located in the \TOOLS\BMGR folder of any repository.
3. Copy the BMGR.EXE file to bootable DOS CD.
4. Create an AUTOEXEC.BAT file (or modify an existing AUTOEXEC.BAT file) to include the following lines:
   ```
   path\bmgr /BS
   echo Remove this CD and restart your computer.
   ```
   where *path* is the path to the folder containing the BMGR.EXE file.

When the computer is started from this CD, the BMGR.EXE program sets up the computer to restart from the service partition on the next reboot.

Appendix. Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
500 Park Offices Drive, Hwy. 54
Research Triangle Park, NC 27709
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO GROUP LTD. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may

vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

The following terms are trademarks of Lenovo in the United States, other countries, or both:

- ImageUltra
- Lenovo
- NetVista
- ThinkCentre
- ThinkPad
- ThinkVantage
- Rapid Restore
- Rescue and Recovery

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

- Cloudscape
- IBM

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

ActionMedia, Intel, LANDesk, MMX, Pentium, and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Index

ThinkVantage™