



BootManage® Administrator User & Reference Manual

Contents

OPERATING SYSTEMS AND OPERATION MODES.....	5
OVERVIEW	6
<i>Operating system installation (with automatic hardware detection)</i>	<i>6</i>
<i>Imaging for installation.....</i>	<i>6</i>
<i>Diskless operation.....</i>	<i>7</i>
<i>Performing one time activities on a Managed PC</i>	<i>7</i>
<i>Diskless Linux network boot</i>	<i>8</i>
<i>Remove an operating system.....</i>	<i>9</i>
<i>Hard Disk Partitioning.....</i>	<i>9</i>
CLIENT INSTALLATION.....	11
<i>Client preparation:.....</i>	<i>12</i>
THE CLIENT PC BOOT PROCEDURE.....	12
<i>Detecting new PCs</i>	<i>12</i>
<i>Releasing a PC.....</i>	<i>12</i>
<i>Hardware information.....</i>	<i>13</i>
<i>Hierarchical groups of the client PCs.....</i>	<i>13</i>
WAYS OF CLIENT INSTALLATIONS (ROLL OUT).....	14
<i>The automatic Roll Out</i>	<i>14</i>
<i>Fully automated Roll-Out.....</i>	<i>14</i>
<i>Selective Roll-Out.....</i>	<i>14</i>
<i>Web Server controlled Roll-Out.....</i>	<i>16</i>
SERVER CONFIGURATION.....	17
SERVER PROPERTIES	18
BACKUP TFTP SHARE	18
WINDOWS DRIVER INTEGRATION.....	19
OVERVIEW	20
WINDOWS DRIVER TYPES.....	20
<i>Plug&Play Drivers.....</i>	<i>20</i>
<i>Boot-time (textmode) mass storage drivers.....</i>	<i>20</i>
<i>Hardware Abstraction Layer (HAL) drivers</i>	<i>20</i>
<i>Executable device drivers.....</i>	<i>20</i>
ADDING WINDOWS DRIVERS	20
WORKING WITH WINDOWS DRIVERS	21
ADVANCED DRIVER SETTINGS.....	21
<i>Driver Description</i>	<i>21</i>
<i>Executable Driver Command line</i>	<i>21</i>
<i>Driver Group Membership.....</i>	<i>22</i>
WORKING WITH DRIVER GROUPS	22
<i>The “DriverHandle” system variable</i>	<i>22</i>
<i>The “DriverGroup” system variable</i>	<i>22</i>
REMOVING A DRIVER.....	23
LINUX OS SYSTEMS.....	24
OVERVIEW	25
DISKLESS LINUX BOOT	25
UNATTENDED LINUX INSTALLATION	26
RED HAT LINUX.....	27
SUSE LINUX	28
CONFIGURATION VARIABLES.....	31
OVERVIEW	32
DEFINE VARIABLES.....	32
<i>Variable Properties.....</i>	<i>33</i>
<i>Value Types.....</i>	<i>33</i>
ASSIGN VARIABLE TO OBJECT	34
SYSTEM VARIABLES	35

USER VARIABLES.....	35
VARIABLE EXAMPLE.....	35
BOOTMANAGE ADMINISTRATOR CUSTOMIZATION.....	38
CHANGE AND CUSTOMIZE LANGUAGE.....	39
<i>Add Additional Languages.....</i>	39
POLLING INTERVALS FOR BACKGROUND ACTIVITIES	40
MEANS OF INDIVIDUALIZATION.....	41
IMPORTANT FILES AND DIRECTORIES	42
<i>Base Directory</i>	42
ADAPTING THE OS.INI	43
<i>OS.INI Structure.....</i>	43
EXTENDING PC PROPERTIES (USER DEFINED MENUS)	46
FIELD TYPES	48
<i>Text Field Type.....</i>	48
<i>Combo Field Type.....</i>	49
<i>Listbox Field Type</i>	50
MULTI SERVER OPERATION.....	51
OVERVIEW	52
IBM AIX NIM SERVER INTEGRATION	53
OVERVIEW	54
NIM OBJECTS	54
ACCESS PERMISSIONS	54
ADD NIM SERVER	55
NIM OBJECT MANAGEMENT	56
THE BOOTMANAGE UTILITIES	58
OVERVIEW	59
PXBOOT/BPBOOT.....	59
<i>Supported Custom DHCP Options (Magic Keywords).....</i>	59
PXUTIL / BPUTIL.....	60
<i>Supported Command Line Parameters</i>	60
<i>Patching text files with individual BOOTP/DHCP Information</i>	60
<i>How does the replacement work ?</i>	61
<i>Transfer BOOTP/DHCP options to environment variables.....</i>	61
BMDRV / BMUTIL32	62
<i>BMUTIL32 Command Line Switches</i>	62
BMFDISK.....	63
<i>Command Line Switches</i>	63
<i>Setup Hard Disk.....</i>	63
<i>Read Hard Disk information.....</i>	64
<i>Modify Partition ID.....</i>	64
BMPCSCAN.....	65
<i>Command Line Switches:</i>	65
INTEGRATING THE BMA IN CUSTOM APPLICATIONS	66
ACTIVE X INTERFACE.....	67
INTERFACE DESCRIPTION OF BMA_SERVER.....	67
SUPPORTED FUNCTIONS.....	68
<i>Show all Client MAC Addresses.....</i>	68
<i>Show all available operating systems of a server.....</i>	68
<i>Show all server names.....</i>	68
<i>Show all Groups.....</i>	68
<i>Show all client properties in an array.....</i>	68
<i>Show all server properties in an array.....</i>	69
<i>Show all operating systems in an array</i>	69
<i>PC Wake-On-LAN.....</i>	70
<i>PC Wake-On-LAN via lookup zone</i>	70

<i>Modify client operating system</i>	70
<i>Reinstall client PC</i>	70
<i>Sample ActiveX call across the network</i>	71
WEBSERVER	72
OVERVIEW OF THE BMA WEBSERVER FUNCTIONS	73
STARTING THE BMA WEB SERVER.....	73
CLIENT MANAGEMENT WITH THE BMA WEB SERVER	73
CONFIGURING THE BMA WEB SERVER	74
BMA WEBSERVER ROLL-OUT.....	74
MODIFYING THE WEB VIEW	77
<i>Precedence rules for individualized files</i>	79
<i>Placeholders within control files</i>	79
BOOT IMAGE MANAGEMENT	80
OVERVIEW	81
DEFINE NEW BOOT IMAGE ENTRY.....	82
<i>Edit & Generate DOS/Win98 Boot Image</i>	83
<i>Import Windows PE Boot Images</i>	84
DELETE EXISTING BOOT IMAGE ENTRY	84
EDIT EXISTING BOOT IMAGE ENTRY.....	84
DISTRIBUTE BOOT IMAGES TO TFTP SERVERS.....	85
CHECK BOOT IMAGE CONSISTENCY ON TFTP SERVERS	85
ASSIGN BOOT IMAGES	85
BARTPE BOOT IMAGES	86
PREREQUISITES	87
AUTOMATING NETWORK INITIALIZATION.....	88
SYNCHRONIZING PLUGIN EXECUTION ORDER.....	89
ADDING 3 RD PARTY DRIVERS TO BARTPE.....	89
<i>Add 3rd party network driver</i>	89
<i>Add 3rd party mass storage driver</i>	89
CLIENT PRESTAGING	90
OVERVIEW	91
SQL SERVER SUPPORT	92
DATABASE MIGRATION	93
<i>SQL Server Properties</i>	93
<i>The Migration Process</i>	93
GENERATING REPORTS	95
THE REPORT GENERATOR DIALOG	96
<i>Report File Formats</i>	96
<i>SQL Queries</i>	96
<i>Templates for HTML Output</i>	96
<i>Generating Rollout & Prestage template files</i>	97
TROUBLESHOOTING	98
<i>PXE-E32: TFTP open timeout</i>	99
<i>PXE-T01: File not Found</i>	99
<i>Configuration Parameter missing</i>	100
<i>The PC can not connect to a Windows 2003 Server</i>	100
<i>Can't connect to network drive</i>	101



Operating Systems and Operation Modes

Overview

The BootManage® Administrator has the capability to administer different operating systems in such a way that a certain operating system can be assigned to each PC.

Before an OS can be used for the client installation that installation files of such OS have to be put once to the basic directory of the BMA. An assistant within the BootManage® Administrator step by step leads through this procedure.

If the automated installation procedure of the desired OS is known to the BMA (e.g. Windows 98, NT 4, 2000, XP, 2003, Vista), the BootManage® Administrator prepares everything in a way that client PCs can be installed immediately.

For flexibility, the formatting of the hard disk and the installation itself have been separated. Each OS has its own directory in the basic directory on the installation server. All corresponding program files and scripts for the installation are being put there.

Operating system installation (with automatic hardware detection)

While the PC loads the boot image the following steps are performed on the hard drive (HD): delete, partition and format the HD. Afterwards the PC connects to the basic directory of the operating systems, updates the hardware information of the PC and then moves to the right OS directory. From here the file INSTALL.BAT is copied to c:\temp, individualizes it and executes it.

Supported operating systems:

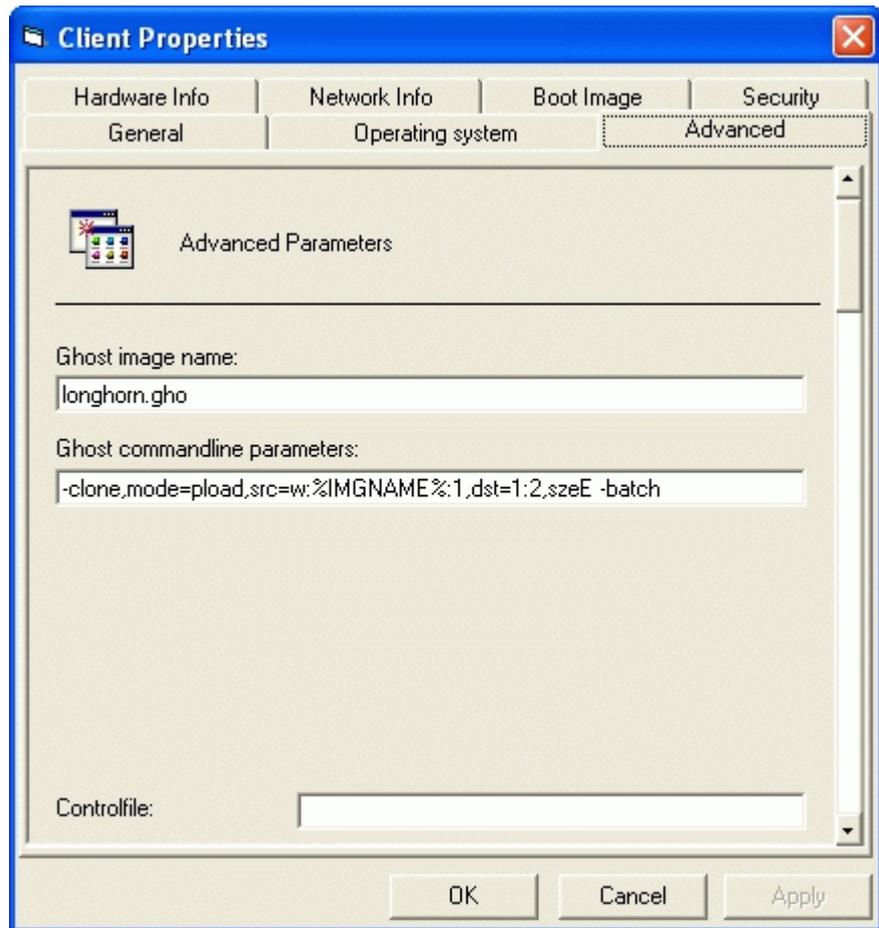
Microsoft Windows 98, NT 4, 2000, XP, 2003, Vista

Imaging for installation

When the PC loads the boot image, the hard drive is deleted, partitioned and formatted. Afterwards the PC connects to the basic directory holding the operating systems to be installed, updates the hardware information of this PC and moves to the operating system directory that has been chosen. From here the file INSTALL.BAT is being copied to c:\temp, individualized and then executed.

The batch files start the imaging program (Ghost or Powerquest). Afterwards the flag partition is created for the network bootstrap program and the PC starts from next time on locally.

It is possible to pass individual parameters to this batch file by specifying these parameters in the advanced client properties (see screenshot).



Supported Imaging tools:

Norton Ghost or Powerquest Drive Image

Diskless operation

While the PC loads the boot image the content of the HD is NOT modified. The PC connects to the basic directory holding the operating systems to be installed, moves to the operating system directory that has been chosen. From here, the file DISKLESS.BAT is being copied to drive A:, individualized and executed.

The batch file may contain programs to be executed now. After the next reboot the PC uses the original boot image again.

Known applications:

- Citrix Clients
- Point of Sale (POS)
- Service-boot which requires a network connection (saving user data)

Performing one time activities on a Managed PC

E.g. data saving

While the PC loads the boot image the content of the HD is NOT modified. The PC connects to the basic directory holding the operating systems to be installed, moves to the operating system directory that has been chosen. From here the file DISKLESS.BAT is being copied to drive A:, individualized and executed.

During execution the batch file generates a message (Your PC has been added) and a flag partition for the network bootstrap program is created. Afterwards the PC starts locally only.

Possible applications:

- Add a managed PC to the data base, WITHOUT new installation.
- Perform one time actions on the managed PC.
- Flashing the BIOS of a PC
- Save data of a managed PC on a server.

Diskless Linux network boot

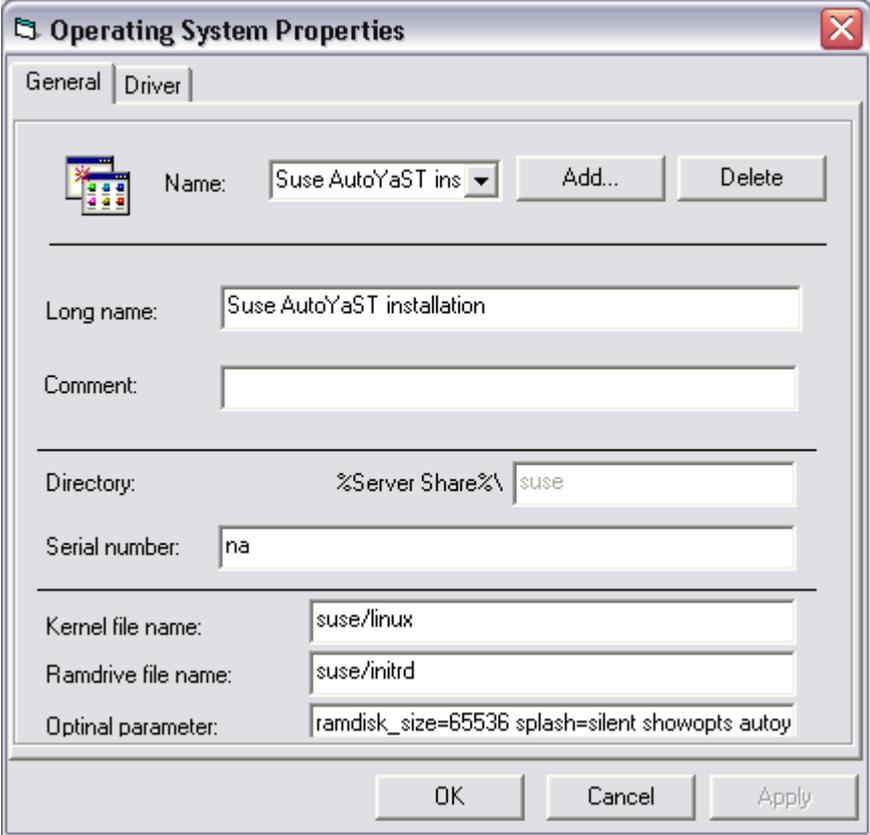
The BootManage Administrator is capable of booting Linux kernels over the network, i.e. in order to provide unattended installations of Linux distributions on client PCs over the network, provide diskless linux based terminal clients, or run linux based offline maintenance operations on PCs that otherwise boot from the local hard disk.

In order to boot a (diskless) Linux system over the network, one needs:

- 1) A Linux kernel that must be network bootable (via PXE). Most Linux distributions already provide such a network bootable kernel.
- 2) A file that contains an initial ram disk for the network boot Linux kernel (the so-called 'initrd')
- 3) (Optional) a set of kernel parameters that are passed to the Linux kernel at boot time and control various aspects of the kernel during the boot procedure.

When you assign a linux based operating system to a client, the client properties view changes and provides three fields that hold the above information (kernel, initrd and kernel parameters).

As the kernel and initrd files are downloaded by the client using TFTP, these files have to be placed into the TFTP share. The BootManage Administrator performs this task for you.



The screenshot shows a Windows-style dialog box titled "Operating System Properties". It has two tabs: "General" (selected) and "Driver". The "General" tab contains the following fields and controls:

- Name:** A dropdown menu showing "Suse AutoYaST ins", with "Add..." and "Delete" buttons to its right.
- Long name:** A text input field containing "Suse AutoYaST installation".
- Comment:** An empty text input field.
- Directory:** A text input field containing "%Server Share%\suse".
- Serial number:** A text input field containing "na".
- Kernel file name:** A text input field containing "suse/linux".
- Ramdrive file name:** A text input field containing "suse/initrd".
- Optimal parameter:** A text input field containing "ramdisk_size=65536 splash=silent showopts autoy".

At the bottom of the dialog are three buttons: "OK", "Cancel", and "Apply".

Remove an operating system

An operating system previously added to the BootManage[®] Administrator for client installation can be removed easily. Open the window „operating system properties“ in the BMA, tick the operating system to be removed, and click „Delete“.

Hard Disk Partitioning

The default partition settings are:

First partition: Full_Disk / Filesystem: NTFS

Second partition: 0

This will create a single system partition of type NTFS that covers the entire available disk space. Any existing data on the hard disk will be overwritten.

To specify the size of the system partition, either change the "First partition" setting to one of the predefined values, or enter your desired size in this box. After doing this, the "Second partition" setting becomes editable. If you leave this value set to its default '0', no data partition is created.

In case you want to create a data partition, either change the "Second partition" setting to one of the predefined values, or enter your desired size in this box.

For the second partition, two special values are available, "REST" and "KEEP". Selecting the "REST" value instructs the BootManage Administrator to create a data partition that takes up all the remaining available hard disk space after the first (system) partition.

For example, three different clients with hard disks of 20GB, 30GB and 40GB should all be installed using a 15GB system partition, and the remaining disk space should be used for data. In this case, all clients can be installed using the same partition setting:

First partition: 15000 MB / Filesystem: NTFS

Second partition: REST / Filesystem: NTFS

As the result, all clients have a 15 GB system partition, but the data partition sizes are 5GB, 15GB, and 25GB, respectively.

During client installation, the BootManage Administrator checks if the selected partition sizes are applicable for the client, and generates an error if the client's hard disk is too small to hold the selected partitions.

Selecting the "KEEP" value instructs the BootManage Administrator to leave an already existing data partition intact, i.e. the contents of a data partition is not altered by the client reinstallation process. Note that this only works when the client's system partition setting has not been changed since the last installation.

Let us assume that the client has been initially installed with the following settings:

First partition: 5000 MB / Filesystem: NTFS

Second partition: 4000 MB / Filesystem: NTFS

Then, it has a 5 GB system partition and a 4 GB data partition.

When the client needs to be reinstalled, but the contents of the data partition should be kept intact, all you need to do is to change the "Second Partition" size to "KEEP" and schedule the reinstallation.

For Windows operating systems, it is recommended to use NTFS as the filesystem type. When selecting "FAT" as the filesystem type, the BootManage Administrator automatically chooses FAT16 for partitions up to 2 GB and FAT32 for partitions greater than 2 GB.

For example, look at the following partitioning scheme:

First partition: 5000 MB / Filesystem: FAT

Second partition: 1800 MB / Filesystem: FAT

For the first (system) partition, the BootManage Administrator automatically selects the FAT32 filesystem type, because the partition size is greater than 2GB. For the second (data) partition, the BootManage Administrator automatically selects the FAT16 filesystem type, because the partition size is smaller than 2GB.

When clicking the "Advanced" button, at this time only a single advanced option can be selected by clicking the checkbox "Fast format the full disk as a single partition". This is the equivalent of specifying the "ExtendOEMPartition" directive within the unattended text file of a scripted Windows installation. When this option is activated, the system partition is automatically extended during NTFS conversion, so that it fills up all the remaining unpartitioned disk space. Please note that this is an intrinsic function of the Windows setup processor, not of the BootManage Administrator, and is mainly kept for backwards compatibility. Only use it when you exactly know what you are doing.



Client Installation

Client preparation:

New client PCs must be configured to be PXE or BootManage TCP/IP BOOT-PROM ready.

The Client PC boot procedure

A BOOT-PROM takes over control of a PC at boot time and receives the necessary configuration data via BOOTP/DHCP over the network.

The TFTP protocol is needed to transfer a Network Bootstrap Program (NBP) to the client, which is then executed, and loads individual configuration data from the server to the client.

The setting of the partition table contains information which is read to decide whether the PC starts from the local hard disk or from the network.

More information can be found in the section which describes the boot process in detail.

Detecting new PCs

The first step to administer a PC is to detect it. When the unknown PC starts for the first time in the network it gets generic configuration data from a DHCP server and then loads a boot image (with the help of a boot loader).

This boot image allows the official recognition of the client, including its hardware PCI and PnP inventory. In the BMA the new PC comes up with a red question mark as member of the \$unknown group. On the PC screen the message below is displayed while the PC waits for the administrator confirmation.



Releasing a PC

The administrator releases a new PC to be automatically installed by providing to this client its individual configuration data, and assigning it to a group of PCs. Now the client has all required data, and continues with the automated unattended installation.

Hardware information

During installation the BootManage Administrator collects the PCI and PnP hardware configuration data of a PC and puts this inventory data into the BMA database.

The BMA uses this information to identify hardware components and to assign the corresponding drivers.

The collected hardware information of a PC can be seen in the BootManage® Administrator. Open the „properties“ of a PC and then tick „Hardware Info“. In the left part of the screen you can see the symbols of the PCI- and PnP components. Clicking on a symbol leads to more detailed corresponding information in the right half of the screen.

Extended WMI based Configuration Information

During the unattended Windows Installation, an additional extended WMI based hardware scan is performed. In order to update the initial PCI/PnP scan information with the extended WMI information for a certain client, wait until the client is completely installed. Then, right-click the client symbol and select “Update Inventory”.

It is possible to update the inventory information for multiple clients at the same time. To do this, select all clients before right-clicking and selecting “Update Inventory”.

The extended WMI based configuration information reflects the Windows “Device Manager” view of a certain client. This allows to see if all hardware devices have been provided with corresponding device drivers. All hardware devices for which no device driver has been found are tagged with a yellow question mark.

The WMI configuration information for a certain client is only integrated in the BMA database after the “Update Inventory” procedure. Note that this affects BMA Reports, i.e. when you want to generate a BMA report based on WMI inventory information, make sure that “Update Inventory” has been executed for each client.

Hierarchical groups of the client PCs

The BootManage® Administrator allows building groups of client PCs in hierarchical structure to keep the overview simple even in large networks. The position of the PCs within the groups may depend on various criteria, e.g.:

- Location (building / floor / room)
- Membership in a department (R&D / marketing / sales)
- Membership in a project group

Below a server symbol groups can be created or deleted. Clicking on a group lets all member PCs of this come up in the right half of the window. The PCs can be shifted around among the groups.

Clicking on a server symbol generates an output on the screen showing all PCs assigned to this server. This is an alternate global view on the hierarchical group tree structure.

All new (first time connected) PCs are automatically assigned by the BootManage® Administrator to the \$unknown group. From there the PCs can a) get the individual configuration data, b) be moved to another group and c) be released for installation.

Ways of client installations (Roll Out)

A client installation can be performed:

- Manually
- By fully automated Roll Out
- By selective Roll-Out

The automatic Roll Out

After a PC has been detected by the BootManage® Administrator for the first time, it appears in the \$unknown group and waits for the administrator to confirm. As soon as the administrator has setup the individual parameters of this PC and has assigned it a group the installation procedure of the client continues.

The assignment of groups and individual data can be automated. This feature of the BootManage® Administrator is named Roll-Out.

Please note: the Roll-Out feature is not an independent Windows service but a functionality of the BootManage® Administrator. The Roll-Out service only works if the BootManage® Administrator Console is running.

Fully automated Roll-Out

This fully automated Roll-Out service provides all unknown (first time) PCs with names consisting of a predefined basic name and an increasing number. The operating system to be installed during the fully automated Roll-Out can be chosen once at the beginning and it the same for all following PCs in this Roll-Out.

Selective Roll-Out

In contradiction to the fully automated Roll-Out the selective Roll-Out only installs Client-PCs automatically which are listed in a specific file. All other PCs remain in the \$unknown group. This specific file contains additional information about each PC to allow an individual installation, which is not the case during the fully automated Roll-Out. The selective Roll-Out is even more powerful in conjunction with time control (Wake up service).



The specific file mentioned above for the selective Roll-Out requires lines with data separated by semicolon (CSV - Character Separated Values).

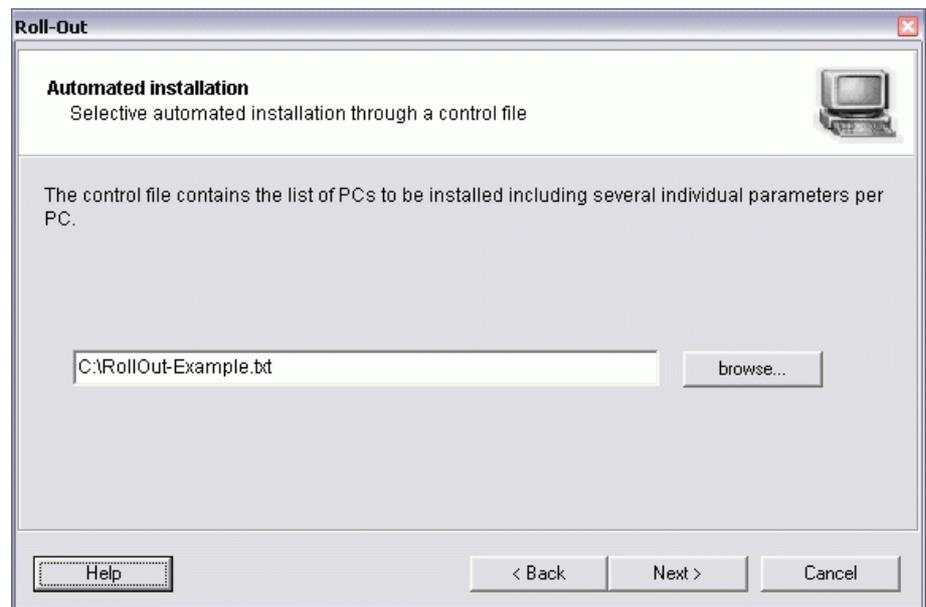
Sample: CSV File

```
MAC ; PC-NAME ; OS-NAME
0F1DEF092201 ; PC001 ; Windows NT 4.0 German
0DD092298Ef1 ; w2kw001 ; Windows 2000
```



The first line of the CSV file determines the type and amount of information that is used to individualize the client installation during selective Roll-Out. The keywords listed in the first line provide a schema for all following lines. For each PC to be installed by selective Roll-Out, you must provide the PC's individual information in a separate line. If one or more settings are not used for a specific PC, you may omit it. Be careful to observe the sequence of the settings – it must match the schema defined in the first line.

In the GUI the automated Roll-Out is located under: Tools/Automated Roll-Out. You are asked to provide a roll-out configuration file. Select your just-created CSV file, and the BootManage Administrator will check it for validity and import the configuration into its database.



Keywords are not case sensitive. A list of all available keywords that can be used within the first (schema) line of the roll-out CSV file is shown below:

Mac	Client MAC address
Name	Client NetBIOS name
FullName	Client full name
InstallOS	Operating system assigned to the client
MainGroup	BMA group to put the client into
FirstPartitionSize	Size of first partition
FileSystem	File system type for first partition
SerialNumber	License key to be used for OS installation
Comment	Comment field
ExtendOEMPartition	Specify if installation partition should be automatically enlarged during automated installation
Organisation	Organization (company name)
BootImage	Name of the boot image file to be used
LinuxKernel	File name of the linux kernel to be used
LinuxRamdrive	File name of the linux initial ramdisk to be used
LinuxParameter	Kernel parameters to be passed to the Linux kernel
Txxx	Custom Tag information (T170 – T194)

Rollout sample files can be found on the BootManage Administrator product CD in the “samples\roll-out” directory. You may want to use these files as templates and modify them to suit your needs.

Web Server controlled Roll-Out

The roll-out function is also implemented in the BMA web server. This has the advantage that one does not need to have a logged-on BMA GUI console running during roll-out operation. Details about this can be found in the chapter about the BMA web server.



Server Configuration

Server Properties

The server properties of the BootManage Administrator hold server dependent parameters, such as server name, login user, login password, along with the locations of the base and TFTP shares.

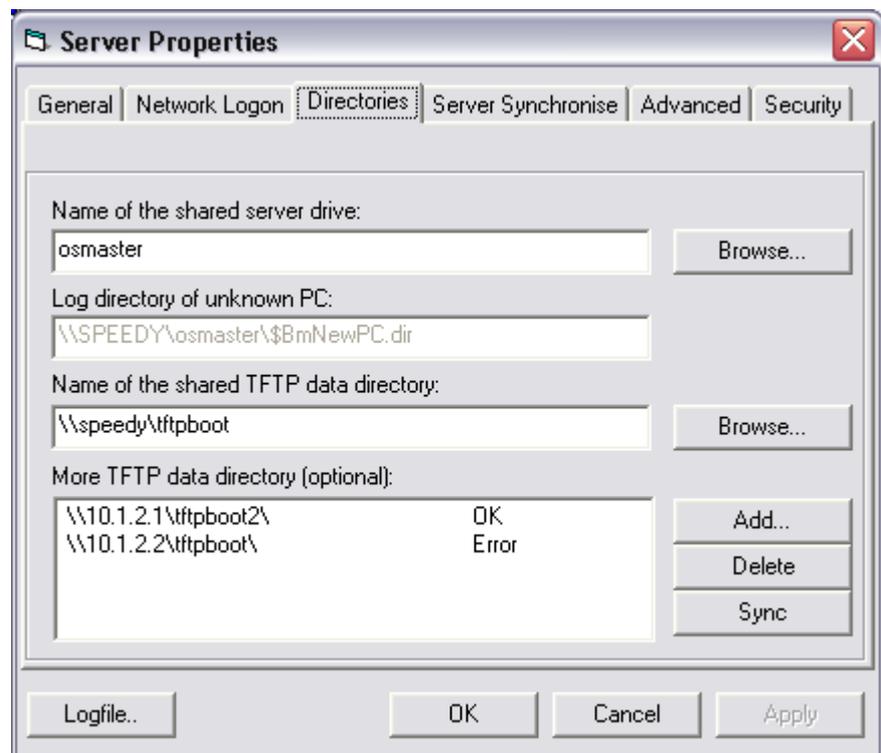
Backup TFTP Share

Each client derives its configuration data from a TFTP server. Normally, each server exports a TFTP share that holds the configuration information for the client PCs. To provide fault tolerance, the BootManage Administrator offers the possibility to define additional TFTP shares. When the properties of a PC are modified, the corresponding configuration information on all TFTP shares is updated. This is also helpful for supporting laptop computers that are moved between different locations of the same company.

In case of a communication error with a backup TFTP server, the corresponding TFTP server is flagged as unsynchronized and must be manually re-synchronized.

To synchronize a TFTP server, proceed as follows:

- In the edit menu of the administration console, select server properties.
- Click the general register and select the server you want to synchronize.
- Click the directories register.
- Highlight the desired backup TFTP share.
- To start the synchronization, click synchronize.





Windows Driver Integration

Overview

The BootManage Administrator allows the easy integration of third party device drivers into an unattended Windows installation. The integration method is based on the Microsoft standard mechanism, but augmented by a driver preselection, grouping and explicit assignment function. So, regarding third party drivers, the BootManage Administrator provides the following features:

- Easy Windows device driver integration by GUI wizard
- Support for Plug&Play drivers
- Support for boot-time (textmode) mass storage drivers
- Support for executable device drivers
- Driver preselection by autodetection and/or explicit assignment
- Possibility of building driver groups

Windows Driver Types

Plug&Play Drivers

Plug&Play drivers are installed during Windows GUI mode setup. In this phase, Windows detects Plug&Play devices, and determines the best suited driver for each detected device from the pool of in-box and third-party drivers.

Boot-time (textmode) mass storage drivers

Boot-time (textmode) mass storage drivers are installed during the early “text mode” phase of Windows setup, where Plug&Play is not yet available. Windows requires a driver for the host adapter to which the system hard disk is attached, i.e. the hard disk that holds the Windows system partition. When this host adapter is not covered by a Windows in-box driver, one needs to add a third party boot-time mass storage driver.

Hardware Abstraction Layer (HAL) drivers

At this time, the BootManage Administrator driver wizard does not support HAL drivers. However, you can integrate HAL drivers in the Microsoft standard way by editing the unattended setup control file, and adding the HAL files to the installation source directory on the installation server.

Executable device drivers

Some third party device drivers can only be installed by running an executable file. If they provide an unattended installation mode via command line switches, the BootManage Administrator supports to install these drivers during the GUIRunOnce phase of the Windows setup process.

Adding Windows Drivers

To add a Windows device driver of any kind to an operating system that is already registered in the BootManage Administrator, use the driver import wizard by selecting “Add a driver” from the “Tools” menu.

In the dialog box that opens, select the operating system to which you want to add the third-party driver, and click the Next button. Note that it only makes sense to add drivers to Windows (NT/2000/XP/2003/Vista) operating systems.

Then, browse to the directory that contains the third party driver files. The BMA driver wizard expects one or more driver INF files in this directory. Note that you will also need a driver INF file for executable device drivers.

You can select a single or multiple driver INF files. This is especially useful for driver collections, e.g. the “Intel Chipset Software Installation Utility” drivers. After selecting the driver INF file(s), click Next to continue.

The BMA driver import wizard now scans all selected INF files and presents all the devices it found in the INF files. You can now select the devices that you want to support with this driver. The “Details” button displays detailed information about a driver. To include all available devices, click the “Mark all” button. Then, click Next.

Now, you can review the PCI ID information for each selected driver and, if desired, also change this information (e.g. to create a more specific driver assignment than the manufacturer provided). Note that at least one set of PCI ID information must be given.

Click the “Finished” button to import the selected driver information, and also import the corresponding driver files. If you want to make sure that all driver files in the driver’s source directory are imported, check the “Copy all files from the driver directory” box. Otherwise, the BMA driver import wizard copies only the files that are referenced in the selected driver INF file(s). Note: When importing executable device drivers, make sure that this box is always checked.

When the BMA driver import wizard detects that the imported driver is a “SCSI Adapter” class driver, it treats this driver as a text mode driver, and checks for the presence of a “txtsetup.oem” file. If such a file is present in the driver source directory, the BMA driver import wizard uses it silently. Otherwise (i.e. if it cannot find the txtsetup.oem file), it prompts for this file. In this case, you must specify the location of the corresponding “txtsetup.oem” file.

Working with Windows Drivers

Once a Windows driver has been added, you can review it in the operating system properties. In the left-hand pane of the BMA console’s tree view, select “Operating Systems”. Then, in the right-hand pane, right-click the operating system to which you added the driver, and select “Properties”. In the properties dialog, click the “Driver” tab.

You can now examine the drivers that have been assigned to this operating system. Each driver directory is represented by a number and an optional description. The OS in-box drivers (i.e. the drivers that are provided as part of the OS) are listed under the special entry “OS Internal”.

By clicking the “Vendor” or “Adapter type” buttons, you can display the available drivers by their vendor name or adapter type.

To display the devices that a driver supports, click the plus sign to the left of the driver number. Highlight a device entry to review its PCI information.

Advanced Driver Settings

To display the advanced driver properties dialog, highlight a numbered driver entry and click the “Advanced” button.

Driver Description

In the “Name” field, you can specify a driver description that appears then next to the driver’s number.

Executable Driver Command line

For executable drivers, you need to specify the command line that executes the driver setup file with its command line parameters for silent (unattended)

execution. To do this, click the “Installation Script” button, and enter the command line in the text file that opens in the editor.

Make sure that you do not include path information! For example, DO NOT specify the driver install command like this:

```
C:\mydrv\driver.exe -silent -x -v
```

Instead, use the following CORRECT example:

```
driver.exe -silent -x -v
```

For executable drivers, make sure that you also check the “Install driver during additional installation script” checkbox.

Driver Group Membership

By default, drivers are autodetected and automatically assigned based on the detected hardware in the target PC. For devices that cannot be autodetected, it is also possible to explicitly assign drivers by using driver groups.

To assign a driver to a driver group, click “Driver Group”, then “Add”, and select the driver group(s) to which you want to assign this driver.

Following, the concept of driver groups and explicit driver assignment is explained in detail

Working with Driver Groups

When integrating third party drivers in an unattended Windows installation, the BMA can preselect drivers using the following mechanisms:

- Autodetect devices and determine the best matching driver
- Explicitly include third-party drivers
- Both autodetect and explicitly include third-party drivers

Note that autodetection only works for PCI devices.

The “DriverHandle” system variable

The “DriverHandle” system variable controls what method is actually used. This variable has three possible values that correspond to the three above listed possibilities), and can be assigned like any other variable to BMA objects.

For example, if you want to specify that, for a certain target PC, drivers should both be autodetected and also explicitly assigned, bind the “DriverHandle” variable to the PC object in the BMA console, and set it to the value “drivers selected with driver group and autodetected drivers”.

Another example: If you want that, for a certain group of target PCs, drivers should only be explicitly assigned, place all these target PCs in a BMA group, bind the “DriverHandle” variable to this PC group, and set it to the value “only drivers selected with driver groups”.

The default behaviour is to autodetect drivers, i.e. when the “DriverHandle” system variable is not specified, drivers are always autodetected.

The “DriverGroup” system variable

It is not possible to explicitly assign individual drivers to BMA objects – this must be done via driver groups instead.

Proceed as follows:

First, define driver groups by providing the corresponding driver group names via the “DriverGroup” system variable. To do this, select “Define Variables” from the “Tools” menu, click on the “@DriverGroup” variable in the left-hand pane, and edit the values in the right-hand pane, so that all possible driver group names are listed. It makes sense to provide driver groups based on special hardware.

Example: Assume that you want to create driver groups for the following hardware models:

- Laptops of type “OnTheRoad 66”
- Desktops of type “OfficeWorkHorse 0815”
- Servers of type “FastAsLightning 3000”

You might want to create three (additional) values for the “@DriverGroup” variable, and name them OnTheRoad_66, OfficeWorkHorse_0815, and FastAsLightning_3000.

Next, we need to assign individual drivers to the different driver groups, so that each driver group represents the collection of drivers that is required for the different machine types. For this purpose, in the properties of each driver, select “Advanced”, and then “Driver Group”. Click “Add”, and select the driver group in which this driver should be added as a member. A single driver can be a member of multiple driver groups.

Having done this, we are now at the point where our driver groups represent the driver portfolio for the corresponding hardware platforms.

The last step is to assign the driver groups to the BMA objects, so that during the unattended installation, the drivers of this group are included. You can assign a driver group to various BMA objects: an individual PC, a BMA group, an operating system, or an installation server. Right-click the object symbol, select “Variables”, then “Add Variables”. Add the “DriverGroup” variable, and select the proper value (which represents the hardware type).

Make sure that you also set the “DriverHandle” variable so that explicit driver assignments are actually performed for this object.

Removing a Driver

When removing a driver, make sure that this driver is no longer needed by any managed client. Otherwise, future client reinstallations may fail.

Therefore, the „Remove Driver“ function is hidden within the BMA console.

To remove a previously installed driver, select the corresponding entry in the operating system’s driver list, right-click it, and select “Delete”.

Note that the driver files will be physically removed from the installation server, and the driver will no longer be available.

Again, before removing a driver, make absolutely sure that it is no longer needed by any client.



Linux OS Systems

Overview

The BootManage® Administrator supports Linux target operating systems in the following ways:

Network Booted Diskless Linux Clients

Diskless boot Linux on client PCs that may or may not be equipped with a local hard disk. This is useful for thin clients, kiosk PCs, training rooms and classrooms, emergency boot, maintenance operations, offline virus scan, etc. The BootManage Administrator supports these types via the “Diskless Linux” OS type.

Completely Network Based Linux Installation

Perform completely network based Linux installations, where not only the Linux distribution installation source is located on a network server, but also the Linux kernel itself is booted from the network via PXE, so that no local media (CD/DVD, floppy diskette, etc.) is needed to initiate the installation process.

Even more, the Linux installation can be completely automated by specifying an “installation script” file as kernel parameter. The Linux kernel downloads this file during the network boot process and uses it as an “unattended answer file” for the installation, so that the installation operates completely automated.

For automated installation, two methods are supported: “Kickstart” for Redhat distributions and “autoyast” for Suse distributions. The BootManage Administrator supports these types via the “Redhat Kickstart Installation” respectively the “Suse Autoyast Installation” OS types.

Diskless Linux Boot

In order to diskless boot a Linux kernel over the network, one needs the following three components:

- The Linux kernel itself which, of course, must be built with support for diskless network boot. Most Linux distributions already provide such a kernel.
- An “initial ram disk” that acts as the root file system for the Linux kernel.
- A set of kernel parameters that control the kernel behaviour at boot time.

For Linux operating system types, the BootManage Administrator allows to specify these three components within the OS properties dialog as follows:

Kernel file name:

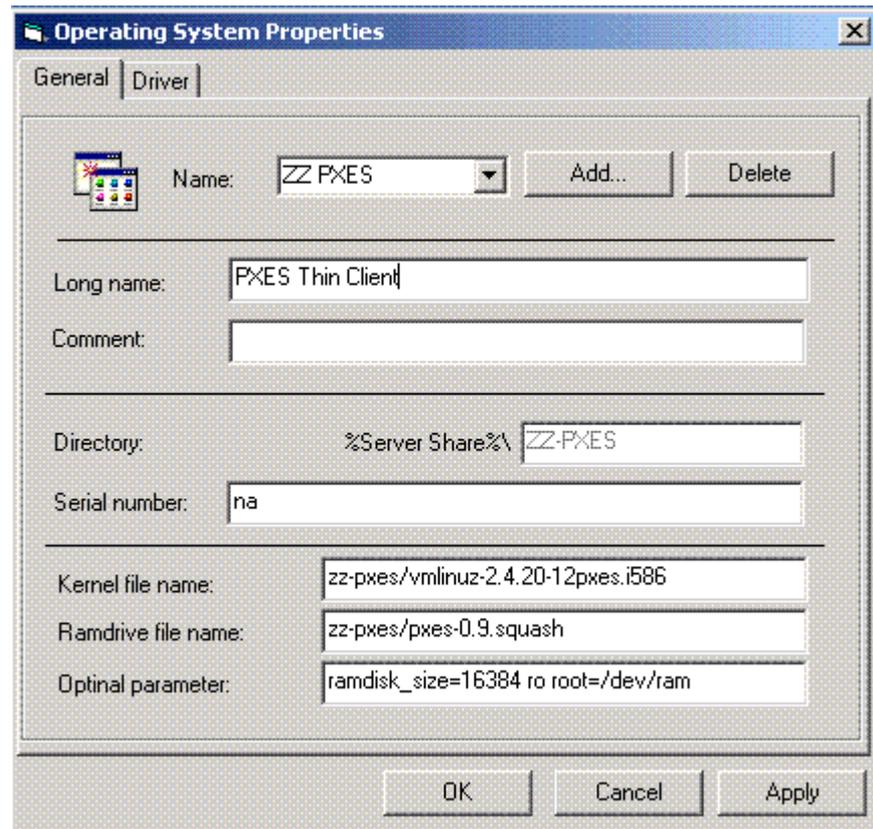
The file that contains the kernel itself, specified as a relative path within the TFTP directory.

Ramdrive file name:

The file that contains the initial ramdisk (initrd), also specified as a relative path within the TFTP directory.

Optional parameter:

The parameters that are passed to the kernel when it is invoked via the boot loader.



If a Linux based operating system does not boot as expected, you should first check whether these settings are correct. Also make sure that the referenced files (kernel and initrd) are present in the corresponding directories.

Unattended Linux Installation

A completely automated, unattended Linux installation differs from a simple diskless boot Linux system in only two ways:

The location and file name of an answer file is provided as an additional kernel parameter. This answer file is not downloaded by the boot loader. Instead, the Linux setup loader downloads the answer file and uses the information within it to perform an automated installation. Depending on the Linux distribution, the Linux setup loader is able to use different methods for downloading the answer file (HTTP, FTP, NFS, etc.).

Also, the Linux distribution to be installed on the clients must be made available on an installation server so that the Linux setup loader can access it using a supported method (HTTP, FTP, NFS, etc.).

After the automated installation is completed, a status ID must be set on the client PC in order to indicate that the next client PC start should take place from the local hard disk. As this status ID is located in the partition table of the client's local hard disk, the automated installation procedure must include a mechanism to perform the proper alteration of the partition table.

Basically, any tool that can set a partition entry's ID value is suited to do the job. However, the BootManage Administrator provides a Linux version of the BMFDISK utility for this purpose.

Red Hat Linux

The Red Hat Linux installation can be automated using “kickstart”, an installation system that is controlled by a simple text based answer file. We assume that you are already familiar with “kickstart”. If you do not know how to create kickstart configuration files and use them to perform unattended installations of Red Hat Linux, please refer to the widely available information on the internet.

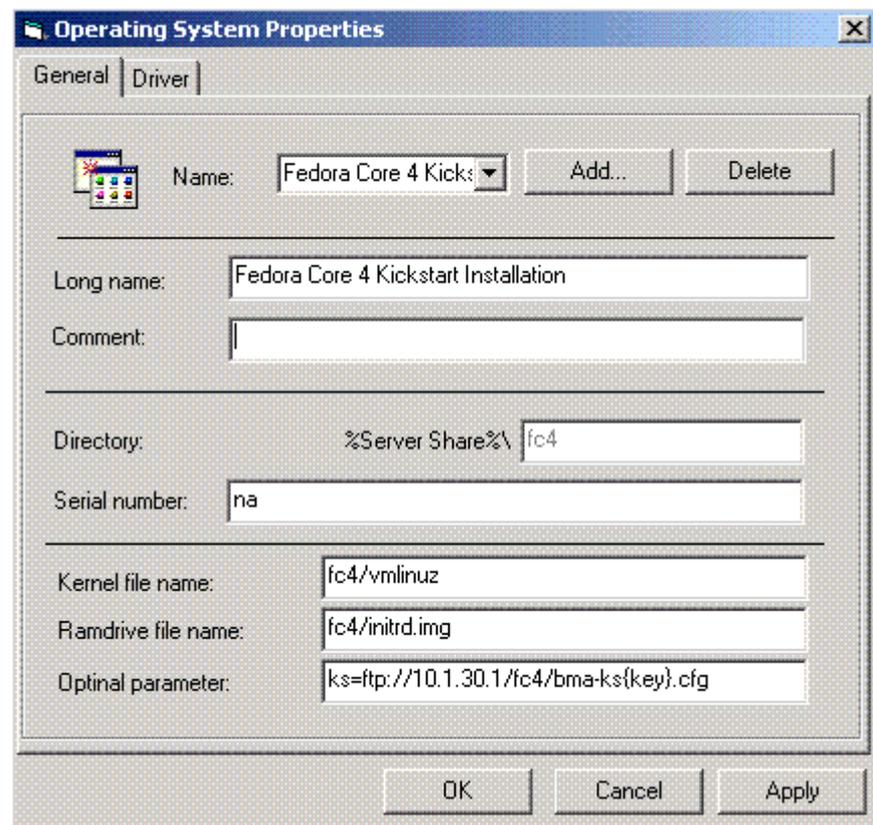
To perform a Red Hat kickstart installation with the BootManage Administrator, please proceed as follows:

Add a new operating system of type “Redhat Kickstart Installation” and give it a short (8.3) as well as a long (descriptive) name. The short name will be used to create subdirectories of both the TFTP and base shares.

A file select box asks for the Linux kernel. The default filename is “vmlinuz”, but you may also change this to suit your needs. The file you specify here will be copied to the above mentioned subdirectory of the TFTP share.

Next, a file select box asks for the initial ram disk file. The default filename is “initrd.img”, but you may also change this to suit your needs. The file you specify here will be copied to the above mentioned subdirectory of the TFTP share. If a file named “initrd.img” is already present in the source directory, it will be automatically and silently copied without displaying a file select box.

Also, as a backup, the kernel and initrd files will be copied to the above mentioned subdirectory of the base share.



The BootManage Administrator fills out the Linux OS properties with sample values (Kernel file name / Ramdrive file name / Optional parameter). Most likely, these sample values need to be adapted manually to reflect your actual configuration.

You may need to make manual adjustments to the path to the kernel and initrd files. Make sure that these path names are correct, and that the kickstart file is correctly specified.

Sample path to kickstart file: “ks=<ftp://10.1.30.1/fc4/bma-ks{key}.cfg>”

(The meaning of the {key} parameter will be discussed later).

In addition, you must manually copy the contents of the Red Hat Linux installation CD / DVD to the installation server.

In order to flag the completion of the kickstart installation to the BootManage Administrator, a status ID must be set in the client’s partition table. For this purpose, you may extend the kickstart control file to fetch and install the bmtools rpm package as part of the unattended installation, and then perform a few custom commands to set the status ID from within the kickstart control file:

```
%post
cd /tmp
wget "http://10.1.4.70/redhat/RPMS/bmtools-1.0.0-1.i386.rpm"
rpm -i bmtools-1.0.0-1.i386.rpm
bmfdisk -o /dev/hda,/dev/sda 3 f2 -f
```

The above script uses the BMFDISK tool to set the partition ID of the last (3) partition on the first hard disk (/dev/hda or /dev/sda) to the value f2.

The bmtools rpm package can be found on the BootManage Administrator CDROM.

In order to being able to trigger the automated reinstallation of a Linux client, you must provide two kickstart files that contain commands to set different status IDs. The above mentioned {key} parameter helps with this. In the example, the file name “bma-ks{key}.cfg” resolves to either “bma-ks0.cfg” or “bma-ks1.cfg”, depending on the internal reinstallation status for this client PC.

So, referring to the above example, you need to provide two kickstart files, “bma-ks0.cfg” and “bma-ks1.cfg”. Within “bma-ks0.cfg”, set the status ID to the value hex f2 or decimal 242, and within “bma-ks1.cfg”, set the status ID to the value hex e2 or decimal 226.

Apart from the different status ID, the two kickstart files should be identical.

Suse Linux

The Suse Linux installation can be automated using “autoyast”, an installation system that is controlled by an XML style, text based answer file. We assume that you are already familiar with “autoyast”. If you do not know how to create autoyast configuration files and use them to perform unattended installations of Suse, please refer to the widely available information on the internet.

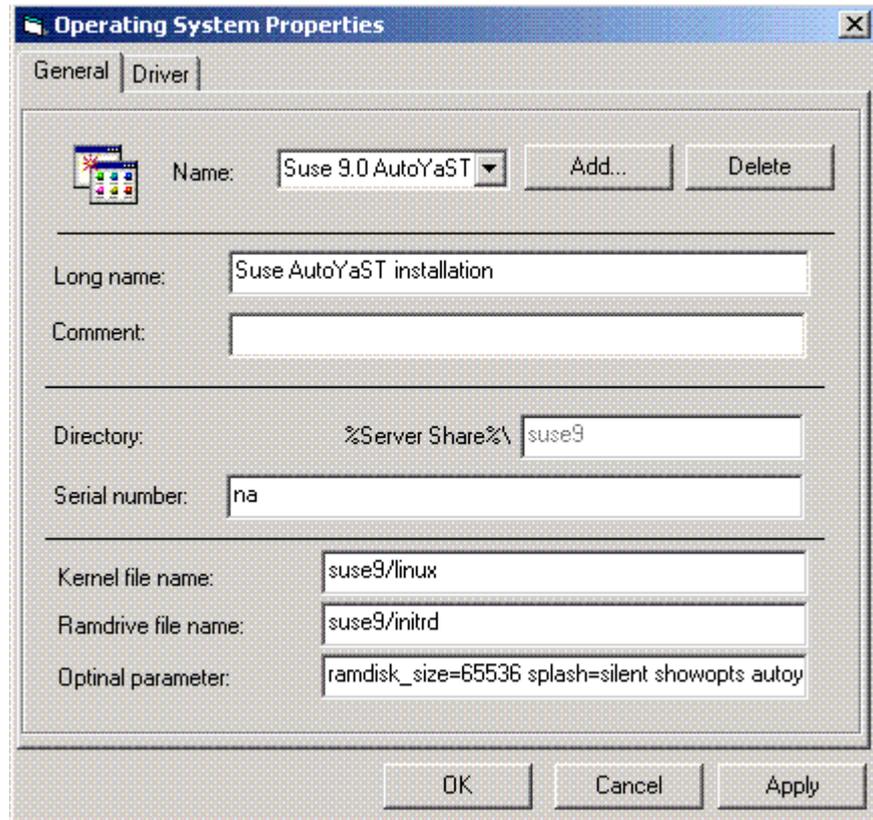
To perform a Suse autoyast installation with the BootManage Administrator, please proceed as follows

Add a new operating system of type “Suse autoyast installation” and give it a short (8.3) as well as a long (descriptive) name. The short name will be used to create subdirectories of both the TFTP and base shares.

A file select box asks for the Linux kernel. The default filename is “linux”, but you may also change this to suit your needs. The file you specify here will be copied to the above mentioned subdirectory of the TFTP share.

A file select box asks for the initial ram disk file. The default filename is “initrd”, but you may also change this to suit your needs. The file you specify here will be copied to the above mentioned subdirectory of the TFTP share. If a file named “initrd” is already present in the source directory, it will be automatically copied without asking you.

Also, the kernel and initrd files will be copied to the above mentioned subdirectory of the base share.



The “Optional parameter” entry is not fully visible in the above screenshot. In our example, it contains the following value:

```
ramdisk_size=65536 splash=silent showopts
autoyast=nfs://10.1.30.1/suse9/bma-ay{key}.xml install=nfs://10.1.30.1/suse9
```

The BootManage Administrator fills out the Linux OS properties with sample values (Kernel file name / Ramdrive file name / Optional parameter). Most likely, these sample values need to be adapted manually to reflect your actual configuration.

Make sure that the path to the kernel and initrd files is correct, and that the installation source directory and autoyast files are correctly specified. As opposed to the Red Hat Kickstart method, Suse Autoyast requires that the installation source is provided as a kernel parameter, and not within the configuration file.

Sample path to installation source: “install=nfs://10.1.30.1/suse9”
Sample path to autoyast files: “autoyast=nfs://10.1.30.1/suse9/bma-ay{key}.xml”

In addition, you must manually copy the contents of the Suse Linux installation CD / DVD to the installation server.

In order to flag the completion of the autoyast installation to the BootManage Administrator, a status ID must be set in the client's partition table. For this purpose, you may use the autoyast-intrinsic partitioning features to set the status ID from within the kickstart control file.

```
<!-- Flag Partition 226=e2 242=f2 -->
<partition>
  <partition_id config:type="integer">242</partition_id>
  <format config:type="boolean">>false</format>
  <partition_nr config:type="integer">1</partition_nr>
  <partition_type>primary</partition_type>
  <size>10mb</size>
</partition>
<!-- Flag Partition END -->
```

In order to being able to trigger the automated reinstallation of a Linux client, you must provide two autoyast files that contain commands to set different status IDs. The above mentioned {key} parameter helps with this. In the example, the file name "bma-ay{key}.xml" resolves to either "bma-ay0.cfg" or "bma-ay1.cfg", depending on the internal reinstallation status for this client PC.

So, referring to the above example, you need to provide two autoyast files, "bma-ay0.cfg" and "bma-ay1.cfg". Within "bma-ay0.cfg", set the status ID to the value hex f2 or decimal 242, and within "bma-ay1.cfg", set the status ID to the value hex e2 or decimal 226.

Apart from the different status ID, the two autoyast files should be identical.



Configuration Variables

Overview

Configuration variables are used to pass information to the client during the unattended setup process. A set of standard variables exists, but one can also define and add custom variables. The variables are hierarchically organized, and can be assigned to various objects:

- Global variables (also called basic variables)
- Operating system variables
- Server variables
- Group variables
- Client variables

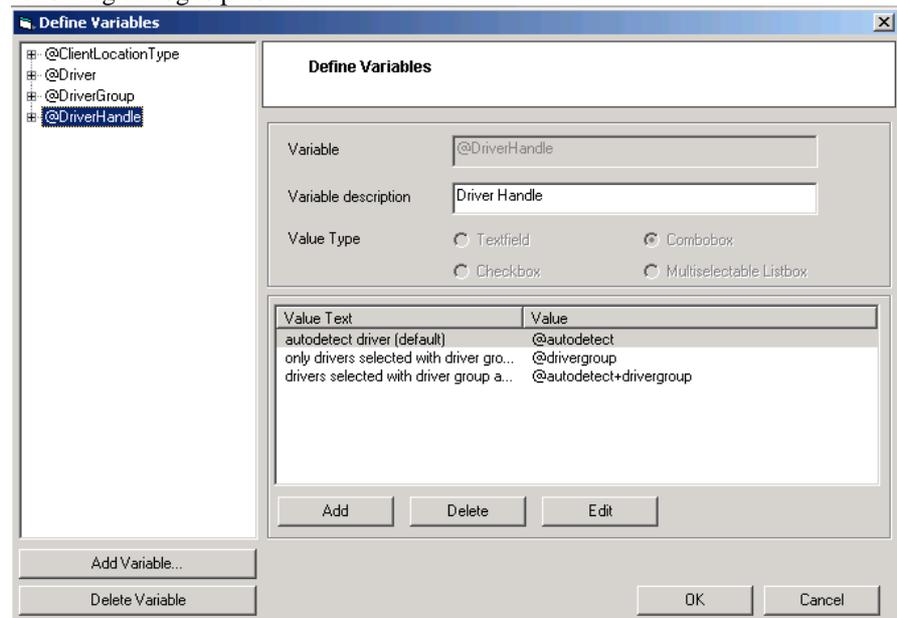
The hierarchical organization allows variable inheritance, i.e. a global variable is automatically assigned to all other objects, and a client that is member in a certain group automatically inherits all variables of that group. The above object list also shows the inheritance paths.

Note that a variable is only inherited from a higher-level object if the variable is not already defined for the object itself. Consider the following example: The PC “ENGR123” is member of the group “ENGINEERING” which, in turn, is member of the server “SRVENG4”. Imagine that the variable “ENGVAR1” is explicitly assigned to the server “SRVENG4”. By inheritance, the variable “ENGVAR1” is also automatically assigned to group “ENGINEERING” and PC “ENGR123”. Then, imagine that the same variable (with a different value) is also explicitly defined for the PC “ENGR123”. Then, the explicit assignment (to the PC) overrides the assignment through inheritance.

An operating system variable is only effective for a client when the corresponding operating system is assigned to that client.

Define Variables

Variables must always be defined before they can be assigned to an object. In order to define a variable, select “Define Variables” from the “Tools” menu. The following dialog is presented:



In the left-hand pane, you can see all defined variables, regardless of their object assignments. System variables are preceded with an ‘@’ sign, are required for BMA internal functions, and cannot be deleted. All variables without a preceding ‘@’ sign are user variables that can be freely defined and deleted.

Variable names are case sensitive, so the variable “MYVAR” is not the same as “MyVar” or “myvar”.

Note that when defining variables, you must not only specify the variable’s name, but also all other properties. The “textfield” value type is the only one that allows entering a specific value when assigning a variable to an object – the values for all other types must be defined here, and can only be selected when assigning a variable to an object.

Variable Properties

Variables have the following properties:

- Variable Name
- Variable Description
- Value Type
- Value Text
- Value

Variable Name

A variable name identifies a certain variable. When the variable name starts with an ‘@’, it is a system variable, otherwise it is a user variable.

Variable Description

This is a free-form text, used to describe what the variable is used for

Value type

The value type defines the way in which the value can be entered via the BMA GUI: as a textfield, a checkbox, a combo box, or a multi-select list box.

Value text

Not the actual value, but a descriptive text that is associated with the value. The value text is especially useful when the actual value is nondescriptive. A good example for this is the timezone which, being a simple number, does not reveal human-readable information about the timezone it represents.

Value

This is the actual value that is used within scripts, configuration files, and other files that are used in the automated setup process.

Value Types

The following value types are available:

- Text field
- Checkbox,
- Combo box
- Multi-select listbox

Text field

This is a free-form text field

Checkbox

This is a selectable field that can only have two states – checked (value text ‘1’) or unchecked (value text ‘0’). The values for the checked and unchecked states can be freely defined.

Combo box

This is a drop-down box that offers a set of possible selections. Only a single selection is possible.

Multi-select listbox

System Variables

System variables are always preceded by an '@' sign, and have a special meaning within the BootManage Administrator. You will find that it is not possible to delete a system variable.

The BMA provides the following system variables:

- @ClientLocationType
- @DriverGroup
- @DriverHandle

@ClientLocationType

Specifies how the client should be handled when relocated within the hierarchical BMA structure.

@DriverGroup

Used to build groups of OEM drivers, which can then be assigned to clients. For detailed information about driver groups, see the corresponding chapter.

@DriverHandle

Controls how the driver is handled with a driver group (e.g. autodetected and/or explicitly assigned).

With the exception of the @DriverGroup variable, you should not try to alter system variables or their predefined values, as they have fixed internal meanings to the BMA.

When you want to add a new driver group, you must do this in the "Define Variables" dialog.

User Variables

User variables are extremely useful when you want to pass configuration information to clients that controls/augments/extends the unattended installation process. The basic idea of user variables is:

- Define the variable in the BMA GUI (Tools – Define Variables), and also define possible values for this variable
- Assign the variable to an object, and select one of the predefined values
- Embed the variable in a script, configuration file, etc. using the placeholder syntax #@VARIABLE@#. For system variables, use the syntax #@@SYSTEM_VARIABLE@#
- When an unattended installation actually takes place, the variable's placeholder is replaced by its value.

Variable Example

The following example illustrates how to use variables. The standard Windows unattended setup file (unattend.txt) contains the following entry:

```
[GuiUnattended]  
  TimeZone=004
```

This sets the computer's time zone to a fixed value of 004, which represents "Pacific Standard Time" or "(GMT-08:00) Pacific Time (US and Canada); Tijuana".

If the computers in a BMA managed environment are located in different timezones, you will want to specify the time zone via the BMA GUI, and you will also want to select the time zone using a descriptive name instead of a nondescriptive number). This is exactly what BMA configuration variables are designed for.

There are three steps to perform:

- 1) Define the variable and its possible values
- 2) Assign the variable to an object, and choose one of the predefined values
- 3) Insert the variable placeholder into your script, configuration file, etc.

Step 1: Define Variable

For our time zone variable, we will use the name “TimeZone”. Note that variable names are case sensitive, so make sure that you enter the name exactly as seen above.

In the BMA console, select “Define Variables” from the “Tools” menu.

Click “Add Variable”, enter the name “TimeZone”, the description “Time zone of the computer”, and the type “Combobox”.

Make sure that the variable “TimeZone” is selected in the left-hand pane, and click the “Add” button.

In the opening dialog box, enter a time zone’s description (value text), and the corresponding number of this time zone (value). Then, click OK.

Again, click the “Add” button to enter additional pairs of time zone descriptions and associated numbers such as:

Value Text: Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna (GMT+01:00)
Value: 110

Value Text: Greenwich Mean Time : Dublin, Edinburgh, Lisbon, London (GMT)
Value: 085

Value Text: Pacific Time (US and Canada) (GMT-08:00)
Value:004

Finally, close the “Define Variables” dialog box by clicking its OK button.

Now, you have defined the “TimeZone” variable, along with its potential values and a descriptive text for each value.

Step 2: Assign Variable to Object

Now that the TimeZone variable is defined, it can be assigned to an object. You may want to provide a default setting, and override this as needed for groups or individual clients.

To provide a global TimeZone default, select “Global Variables” from the “File” menu, and click “Assign Variables”. In the opening dialog box, click “Add Variable”, select “Time zone of the computer” from the drop-down list, and then click “Add”. Now, the variable appears in the list, but it has an empty value.

Next, double-click on the value field next to the variable name. From the drop-down list, select the desired global value and click OK.

You have now assigned a global variable which is inherited by any managed server, group, or client. To verify this, right-click on any server, group, or client symbol, select Properties, and then click the Variables tab. By default, only directly assigned variables are displayed. To display also the inherited variables, click “Inherited Variables”, and you will see the TimeZone.

In order to override the global default time zone, and set a different time zone for e.g. a group, simply assign the same variable to the group. An explicitly assigned variable will always override an inherited variable with the same name. Note that

the explicitly assigned variable will itself act as a new default for all child objects.

Step 3: Insert Variable Placeholder in File

Open the operating system's unattended installation text file with a text editor and locate the following section:

```
[GuiUnattended]
    TimeZone=004
```

Replace the fixed value 004 with the placeholder for the TimeZone variable, which is #@TimeZone@#, as follows:

```
[GuiUnattended]
    TimeZone=#@TimeZone@#
```

The process of replacing a variable's placeholder by its value is called "patching". Any file that contains one or more variable placeholders must be patched before it can be used in the unattended installation process. The BMA standard script and configuration files are already patched, but if you add new script or configuration files, you must make sure that your files are patched as well by running the bmpatch/bmpatch32/bmpatch64 on the file.

To see how this is achieved, examine the BMA_WINPE_PROJECT.CMD file in the operating system base directory.



BootManage Administrator Customization

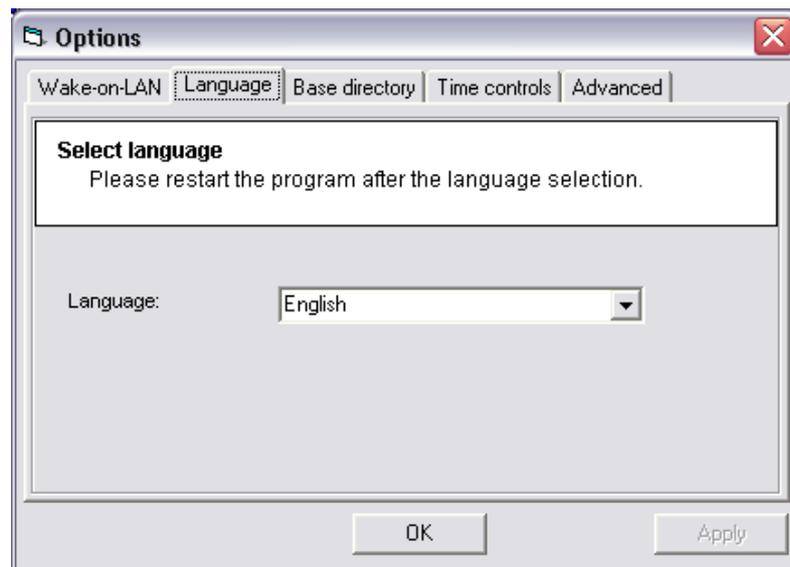
Change and Customize Language

Both the menus and online help of the BootManage Administrator are available in multiple languages.

To change the language, proceed as follows:

- From the tools menu, select options.
- In the options window, click the language register.
- Select the desired language and click accept.

To activate the new language, restart the administrator console.



Add Additional Languages

The button and menu texts as well as all dialog messages of the BootManage Administrator originate from language files that are identified within the configuration file, BMADMIN.INI.

```
[Language]
Default=German
German="%app.path%\msg_de.dat"
English="%app.path%\msg_us.dat"
L01=German
L02=English
```

The BootManage Administrator comes with two language files, MSG_DE.DAT for German and MSG_US.DAT for English text.

Polling Intervals for Background Activities

The BootManage Administrator is able to execute certain activities in the background:

Automated Roll-out:

The BootManage Administrator checks if new clients appear in the \$unknown group, for which a roll-out was scheduled. For each new client, the roll-out is executed.

Time Control Service:

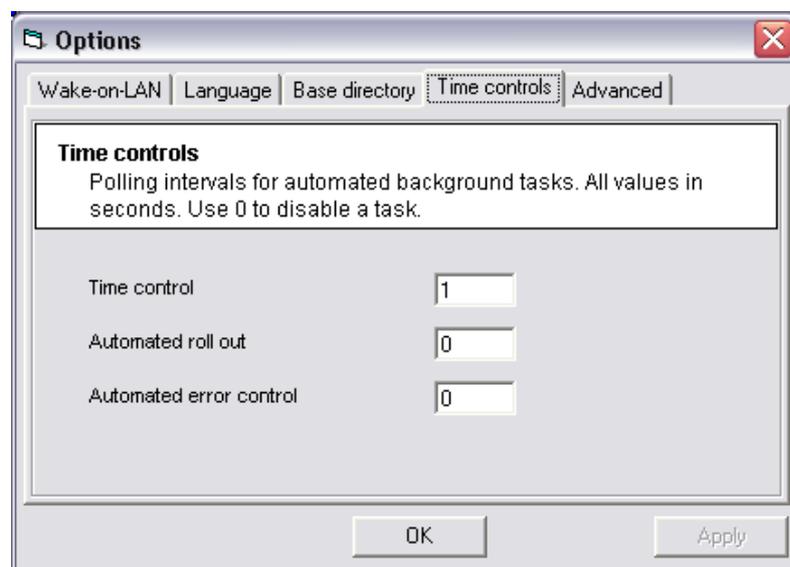
The BootManage Administrator checks the order list in the time control service configuration and executes the orders at the preconfigured times. This includes the time-scheduled installation as well as remote wakeup/shutdown of client PCs via remote at preconfigured times.

Automated Error Detection:

The BootManage Administrator checks if errors have occurred during client installation and generates corresponding messages.

The polling intervals for all background services are defined as follows:

- In the tools menu of the administration console, select options.
- Click the time control register.
- For each service, enter a value in seconds to define the polling interval for the corresponding service. The value 0 deactivates a service.
- Click Apply to activate and save the settings.





Means of Individualization

Important Files and Directories

The BootManage Administrator operates file oriented. The log files are in plain text format and can be extended according to personal requirements.

Base Directory

`\$BmNewPC.dir`

`\$BmNewPC.dir[MAC].chi`

In this directory, new (unknown) PCs store their hardware information. As soon as a file with the naming convention `<last_8_digits_of_MAC_address>.chi` is created in this directory, the corresponding PC icon appears in the \$unknown group of the administration console

`\$BmNewPC.dir\log`

`\$BmNewPC.dir\log\main.log`

`\$BmNewPC.dir\log\[mac].log`

Logging information is stored in this directory. The server log file is named MAIN.LOG. Each PC automatically creates a log file with the naming convention `<last_8_digits_of_MAC_address>.LOG`.

`\$BmNewPC.dir\log\[MAC].lck`

If a PC's network adapter type is not supported, the PC creates a lock file. This file allows restarting the PC from the administration console.

`\$BmNewPC.dir\recent\`

`\$BmNewPC.dir\recent\[mac].chi`

Newly installed PCs store their hardware information in this directory.

Adapting the OS.INI

The OS.INI configuration file contains all information regarding the available operating systems and operation modes that the BootManage Administrator manages.

OS.INI Structure

The section [SupportedOS] contains the section names of all operating systems that the BootManage Administrator supports. This section controls which operating systems are visible.

```
[SupportedOS]
w2k_pro=1
w2k_asrv=1
xp_home=1
xp_pro=1
w3k_w=1
```

Each operating system section contains the following information:

Name:

Operating system name and operation mode

OSDestination:

Target directory for the operating system installation source

Version:

Internal version number

Typ:

1 = create service partition.
2 = do not modify hard disk partition table.

Language:

The operating system installation only supports this language (restrictive language). This function is provided for installations that require a language specific unattend.txt file (e.g. Windows NT 4).

Lang:

Language abbreviation (us or ger), 3 characters maximum.

Common:

When an additional operating system is added to the BootManage Administrator, these files are copied to the target directory.

Path[-9]:

When extending the BootManage Administrator with an additional operating system, these files are copied to the operating system's target directory.

MinPartSize:

Minimum partition size. A value of 0 prevents the partitioning and formatting of the hard disk. From within the boot image, the client directly connects to the base share and executes the DISKLESS.BAT batch file. A value greater than 0 denominates the default partition size value.

SupportedFilesystem

This value controls what file systems appear under „supported file systems“.

CopyFiles:

This entry holds the names of files that should be copied in addition to the default files.

SerialNumber:

The serial number that should be used for the operating system installation.

TSF[0-9]: (Test Search File)

This entry points to a section with conditional entries that must be fulfilled before the installation files can be copied.

Example:

```
[w3k_s.xx]
FileName=ProdSpec.ini
Section=Product Specification
key=Product
Value=Windows Server 2003, Standard Edition
```

The BootManage Administrator looks for the Prodspec.ini file, in which a section “Product Specification“ with the entry „Product= Windows Server 2003, Standard Edition“ must be present. This function guarantees that the correct operating system installation source has been selected.

Advanced

This entry points to a section with additional parameters which are shown in the extended client properties. Using this extension, additional customized parameters can be integrated in the BootManage Administrator menus. For detailed information, please refer to „Extending the PC properties“

Example of an operating system section

```
; Windows 2003 Server, Standard Edition all languages
; -----
;
[w3k_s]
Name=Windows Server 2003, Standard Edition
OsDestination="i386"
Version=1.0
Typ=1
Lang=al
Language=
Common=%app.path%\patch_OS\common
Path=%app.path%\patch_OS\w3k.drv
Path1=%app.path%\patch_OS\w3k_s
MinPartSize=2000
SupportedFilesystem="ConvertNTFS; LeaveAlone"
CopyFiles=
Serialnumber=
TSF0=w3k_s.xx
Advanced=w3k_s_Advanced

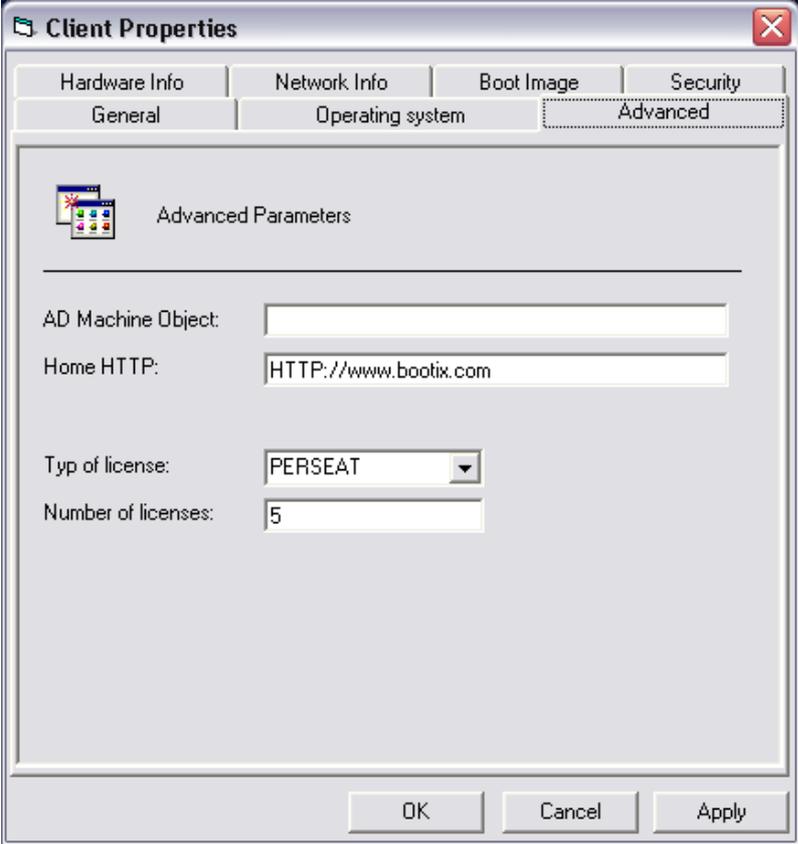
[w3k_s.xx]
FileName=ProdSpec.ini
Section=Product Specification
key=Product
Value=Windows Server 2003, Standard Edition

[w3k_s_Advanced]
```

Extending PC Properties (User Defined Menus)

The PC properties within the BootManage Administrator are extendable. Each PC can be assigned additional parameters that are available during the unattended installation and also under the installed operating system. These additional properties are controlled through the OS.INI file.

Standard View:



The screenshot shows the 'Client Properties' dialog box with the 'Advanced' tab selected. The 'Advanced Parameters' section contains the following fields:

AD Machine Object:	<input type="text"/>
Home HTTP:	<input type="text" value="HTTP://www.bootix.com"/>
Type of license:	<input type="text" value="PERSEAT"/>
Number of licenses:	<input type="text" value="5"/>

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Apply'.

Customized View:

The screenshot shows the 'Client Properties' dialog box with the 'Advanced' tab selected under the 'Operating system' section. The fields are as follows:

IP Address:	10.1.2.144		
Subnet Mask:	255.255.0.0		
Gateways:	10.1.0.254		
DNS Server:	YES (dropdown) 10.1.0.1		
DNS Suffix:	bootix.com		
DNS Domain:	bootix.com		
WINS Server:	YES (dropdown) 10.1.0.1		
Profile Dir:	d:\profiles		
AD Machine Object:	OU = CLIENTS, OU = DETF, OU = STANDORTE/		
Display Color:	16	V Refresh:	70
X Resolution:	1024	Y Resolution:	768

Buttons at the bottom: OK, Cancel, Apply.

In the advanced entry of the operating system section, one can extend the PC property sheets. This entry must point to a section that contains the assignments of the individual parameters.

Example:

```
[w3k_s_Advanced]
T170=AD_MachineOU
T171=DEFAULT_HTTP
T172=SRV_SELECT_LICENSE
T173=NUMBER_OF_LIC
```

The options T170-T194 are available as dynamic keys and can be used later from within DOS and Windows.

Each key points to a section with information about the corresponding field type. The following field types are available:

- Text
- Combo
- ListBox

Field Types

Text Field Type

```
[AD_MachineOU]
Typ=text
default=""
Text="cpMACHINEOBJECTOU"
TextTop=1300
TextLeft=200
Top=1280
Left=2000
Width=3800
Height=195
```

Typ= text

Defines the field type as text.

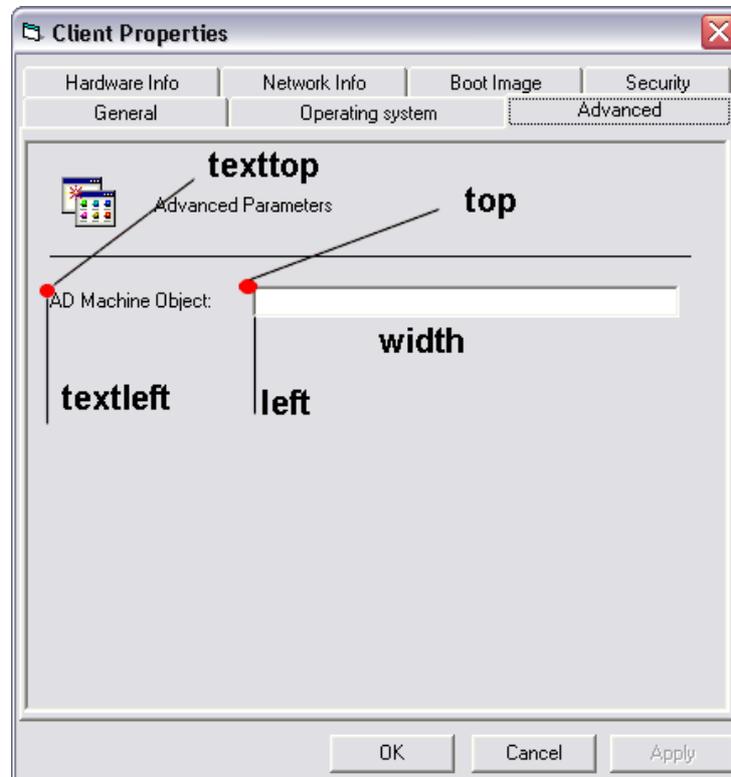
Default=""

Specifies the default value of the field.

Text="cpMACHINEOBJECTOU"

To support multiple languages, the text is first looked up in the language file. If it is not found there, then it is shown as defined in the field type definition.

The additional properties describe the position of text and input field.



Combo Field Type

```
[ SRV_SELECT_LICENSE ]
Typ=Combo
Text="cpLicense"
Default="PERSEAT"
Combo=SRV_LICENSE_DATA
TextTop=2500
TextLeft=200
Top=2480
Left=2000
Width=1800
Height=0

[ SRV_LICENSE_DATA ]
PERSEAT
PERSERVER
```

Typ= Combo

Defines the field type as combo box.

Default="PERSEAT"

Specifies the default value of the field.

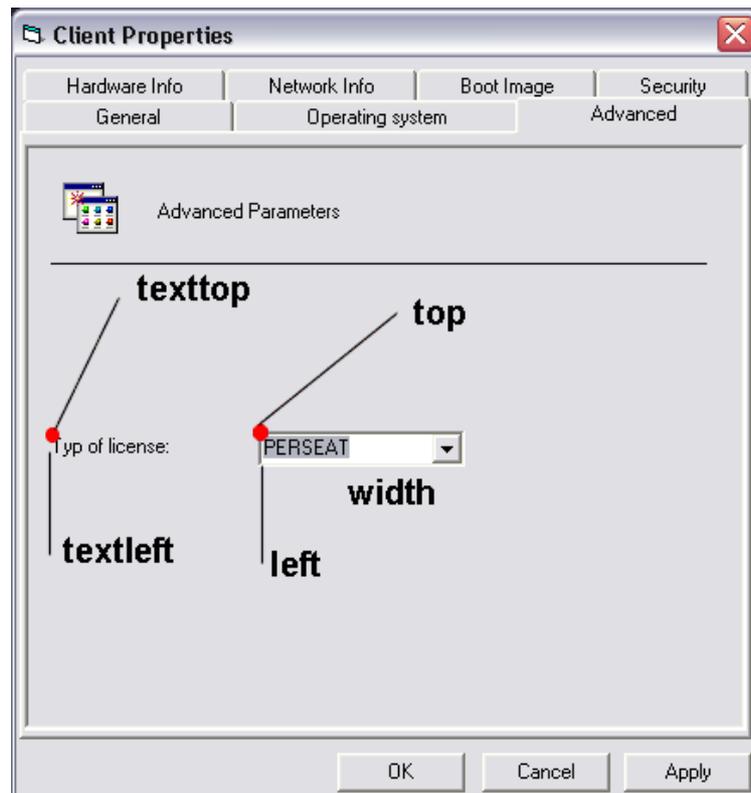
Text="cpLicense"

To support multiple languages, the text is first looked up in the language file. If the text is not found there, it is displayed as specified in the field type definition.

Combo=SRV_LICENSE_DATA

This is a pointer to a section that holds the combo box data.

The remaining keys describe the positions of the text and input fields.



Listbox Field Type

```
[Software]
Typ=ListBox
Text="cpSOFTWARE"
Text2="cpSELECT"
ListBox=ListBoxExample01
TextTop=2200
TextLeft=200
TextTop2=2200
TextLeft2=3250
Top=2780
Left=200
Width=2535
Height=2400

[ListBoxExample01]
01=Acrobat Reader 5.1
02=Windows 2000 Support Tools Setup Databa
03=ActivePerl 5.8.0.805
P4=Open Office 1.0.1
```

Typ= ListBox

Defined the field type as list box

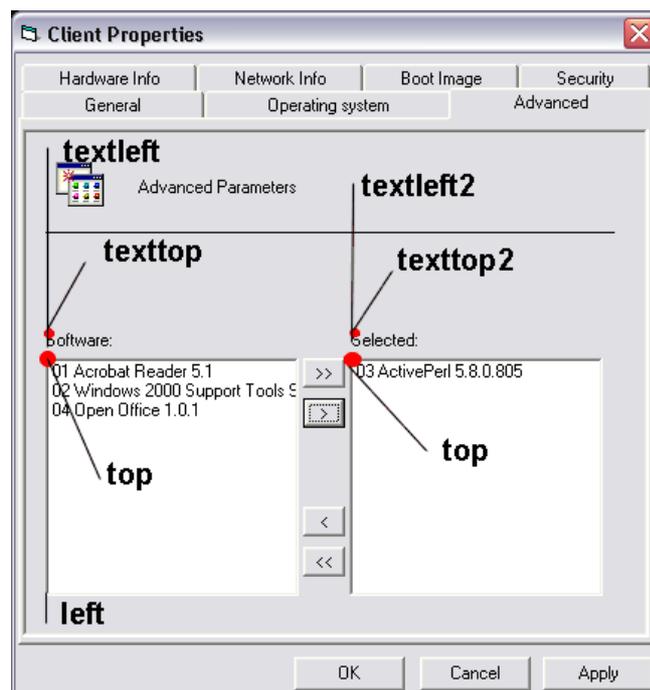
Text="cpSoftware" Text2="cpSelect"

To support multiple languages, the text is first looked up in the language file. If the text is not found there, it is displayed as specified in the field type definition.

ListBox=ListBoxExample01

This is a pointer to a section that holds the list box data. Within a list box entry, values are separated with a semicolon, e.g.:
for Acrobat und Open Office = ;01;P4;

The remaining keys describe the positions of the text and list box fields.





Multi Server Operation

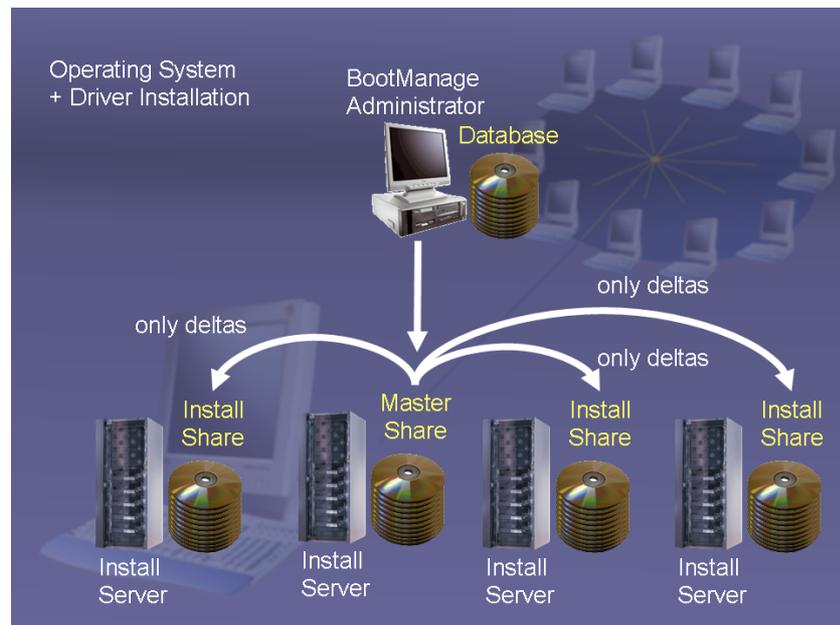
Overview

The BootManage Administrator Multi Server Support allows implementing:

- Redundant and fault tolerant installation servers
- Load-balancing during simultaneous installation of multiple client PCs
- Client management across network borders, subnets and WANs.
- Local installation servers with central management.

If case of an installation server failure, another installation server can take over seamlessly. The BootManage Administrator allows combining fault tolerance with load balancing. Distributed components are centrally managed and synchronized. For synchronization, data is transferred in a compressed format to reduce network load.

Using Multi Server Operation, the BootManage Administrator supports the management of large and distributed networks (also across WAN links). All managed data is available on multiple servers and is synchronized from a central master server. This offers load distribution along with enhanced fault tolerance. During synchronization, only modified data is exchanged in order to minimize network load.



To activate Multi Server Operation, proceed as follows:

- From the tools menu of the administration console, select options.
- Click the extended register.
- Select activate multi server support.
- Click accept to activate and save your settings.



IBM AIX NIM Server Integration

Overview

Starting with version 7.5, the BootManage Administrator supports the integrated and transparent management of IBM AIX NIM (Network Installation Management) servers. These are displayed in the BMA management console like BMA servers. The objects that are managed by NIM server appear grouped by object type, and can be graphically managed. Each action directed to a NIM object is transparently processed by the BMA in the background, converted into a NIM directive and sent to the NIM server. The reply is also evaluated, processed and presented in the BMA GUI.

NIM servers can be added to and removed from a running BMA environment.

NIM Objects

The available NIM objects are:

lppsource

An lppsource is a depot that contains the AIX installation filesets. This is equivalent to the contents of the AIX operating system CDs. An lppsource need not contain the entire contents of the AIX-CDs, and it can also contain additional filesets that are not present on the AIX CDs.

SPOT (Shared Product Object Tree)

A SPOT is a mini filesystem that is mounted on the NIM server during the installation of a NIM client. Normally, a SPOT is created from an lppsource.

Client

A NIM client is an IBM AIX machine that is managed via NIM. This can also be an AIX server or an LPAR (Logical Partitions on IBM pSeries Servers). The term „client“ refers to the role of the managed machine in the NIM system, i.e. such a client is managed via NIM and can be restarted, reinstalled or updated. An LPAR or a server has a specific server platform that must be specified when the client is defined.

User

A user account that exists on the NIM server.

Group

A group account that exists on the NIM server.

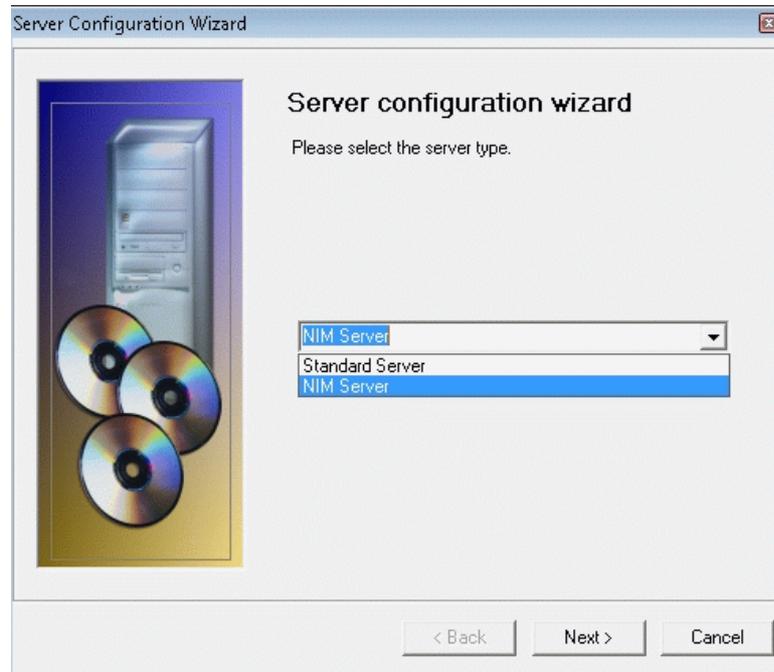
Access Permissions

Before a BMA administrator can manage a NIM server and its objects, he needs to have the execute right for this server.

In order to execute a command on a NIM server, the username and password of an existing NIM administrator user must be specified when adding the NIM server. This user must already exist on the NIM server, but it should NOT be the NIM server's root user. Instead, use a user account that was specially designed for management of the NIM server, e.g. „nimadmin“.

Add NIM Server

To add a new NIM server, proceed exactly as when adding a new standard BMA server. In the new dropdown select box, you can now select whether you want to add a NIM- or a standard BMA server.



After a descriptive name for the new NIM server has been specified, the following configuration dialog is displayed. Here, the parameters of the new server must be specified.



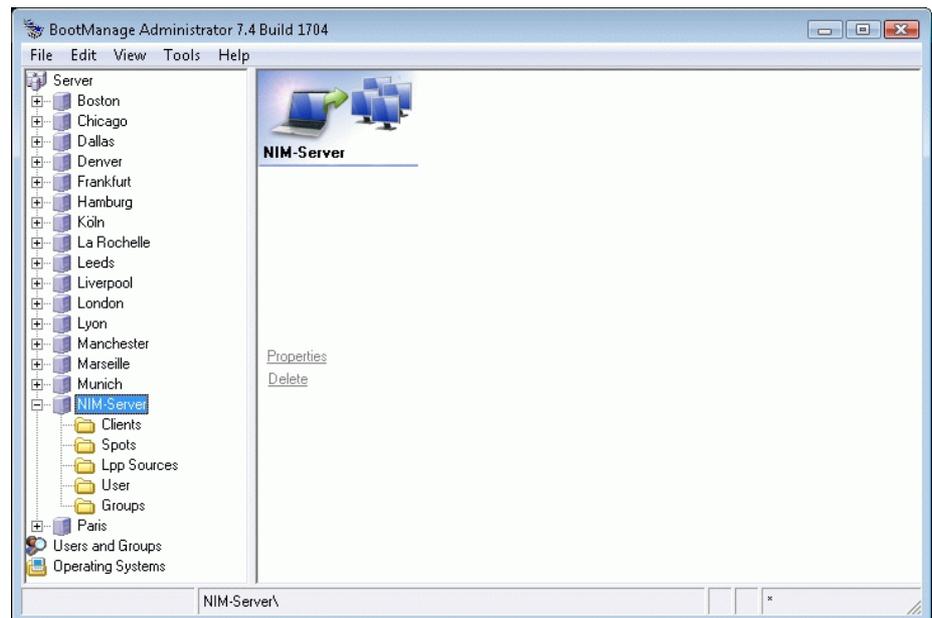
As opposed to the previously entered descriptive name, here the server's (DNS resolvable) hostname or its IP address must be entered. The user name and password must match the credentials of a user account on the NIM server that has the permission to manage the NIM server.

When the configuration is finished, a test is performed whether the NIM server can be contacted via the given hostname or IP address. Also, it is checked whether the given username and password can be used to issue management commands to the NIM server.

During these tests, it is determined how many NIM clients the NIM server actually manages. Regarding BMA licensing, each NIM client is treated like a standard BMA client, and so requires a BMA client license.

If, for example, you want to add a NIM server that already manages 30 NIM clients, then this will require 30 BMA licenses.

After the NIM server has been added, its manageable objects appear in the BMA tree view:



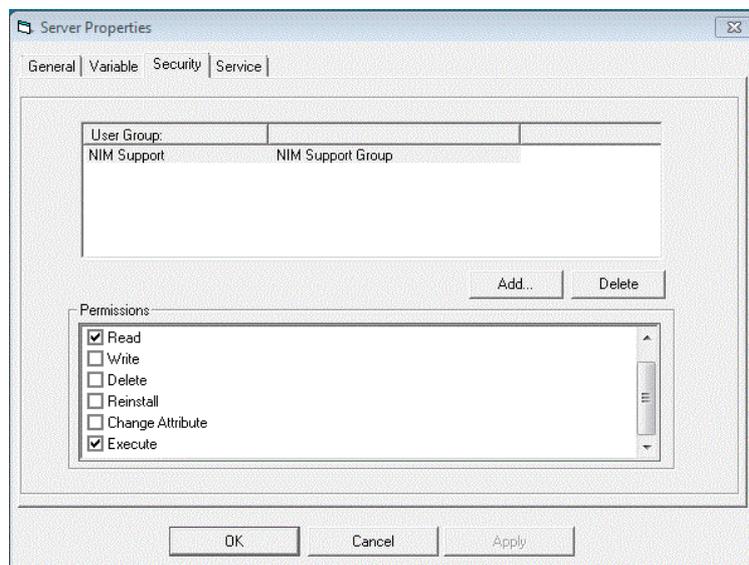
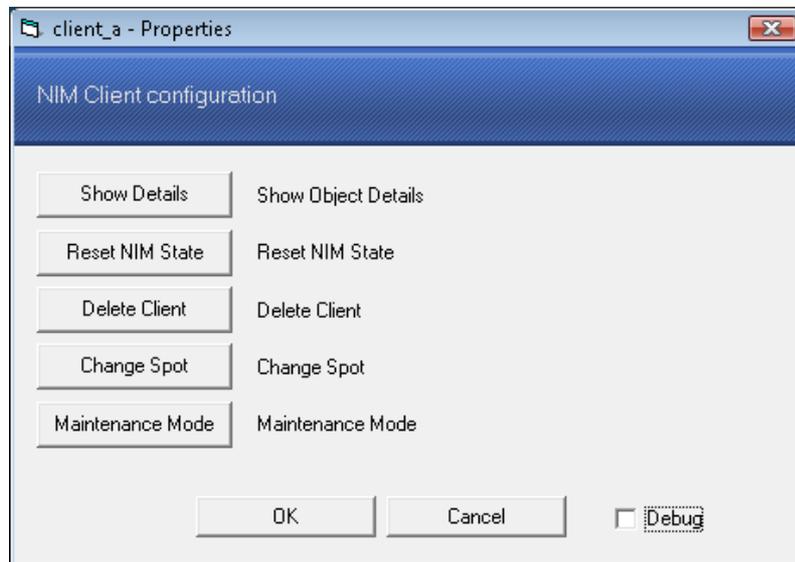
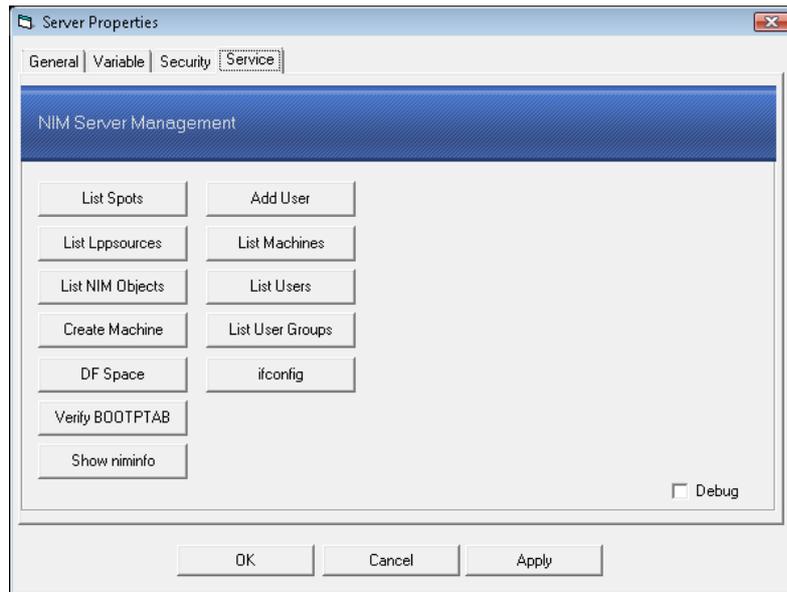
NIM Object Management

For the management of NIM objects, the BootManage Administrator uses the new service control architecture. Each action that can be applied to an object is represented by a button in the object's property dialog. When a specific button is clicked, some additional parameters may be queried, and then the predefined action is taken in the background. The results of the action are processed and displayed in the BMA GUI.

Actions for Clients, Spots, Lpp Sources, Users and Groups can be directly executed from the object's property dialog.

Actions for NIM server itself can be executed from the „Service“ tab in the NIM server's property dialog.

Before a BMA administrator can execute an action on a NIM server, he must have the execute right for this NIM server. This can be specified in the „Security“ tab of the NIM server's property dialog.





The BootManage Utilities

Overview

The BootManage Administrator is based on the BootManage utilities for PXE and TCP/IP BOOT-PROMs. All processes that make use of these utilities can be modified and extended.

PXBOOT/BPBOOT

Instead of a standard boot image, the network bootstrap program (NBP) is loaded and reads status information from the local hard disk's partition table. Based on these settings, the NBP decides whether to boot from the local hard disk or from the network.

The NBP provides special functions to:

- display verbose/debug information (PxDbG=03;)
- download additional custom BOOTP/DHCP options via option files
- provide workarounds for faulty PXE or BIOS codes

Supported Custom DHCP Options (Magic Keywords)

The following magic keywords are understood by the NBP and can be used within custom DHCP options to control the NBP behaviour:

PxSrV	TFTP server IP address
PxSuB	Subnet mask
PxRoU	Router/Gateway IP address
PxOpT	Download global and/or individual user variables from ASCII files
PxInS	Enable procedures for unattended installation
PxDiS	Allow user to request network boot
PxBoO	Define the boot device that PXBOOT should transfer control to
PxFiL	Specify boot image base filename
PxImG	Specify boot image full path and filename
PxRaM	Specify the memory location of the RAM disk
PxFbM	Specify the size of free base memory
PxReS	Local boot debugging / workaround
PxDbG	Display diagnostic information

PXUTIL / BPUTIL

These utilities provide access to the PC's assigned configuration information during network boot and allow patching text based and binary files with this configuration information.

Supported Command Line Parameters

-a fnam	Patch the BOOTP reply into ASCII file.
-b tnum fnam	Patch the BOOTP reply into binary file.
-c	Display the TCP/IP BOOT-PROM configuration
-C	Set ERRORLEVEL if the PC was not booted over the network.
-d dnum	Checksum the diskette drive.
-e	Reboot the PC via System BIOS call.
-E	Reboot the PC via Ctrl - Alt - Del
-f	Fix MS-DOS memory allocation for use with EMM386.
-h	Install the XMS handler entry point.
-i fnam	Include a file within BOOTP reply to extend the BOOTP reply block.
-i [gip] sip fn	Retrieve a file via TFTP and include it within the BOOTP reply to extend the BOOTP reply block.
-m	Display MS-DOS memory control blocks.
-o drv	Set the boot drive
-p fnam	Same as -a but uses a space to separate IP addresses.
-r	Restore TCP/IP BOOT-PROM memory and diskette drive A:.
-s [tag [tags]]	Display all tags from the BOOTP reply. If one or more tags are specified, display only those tags.
-S	Display only those tags from the BOOTP reply that have a value assigned.
-t [gip] sip src dst	TFTP transfer file src to file dst from the network server with IP address sip.
-u r w rw u pwd	Protect or unprotect diskette drives.
-v addr32	Relocate the RAM disk to memory address addr32.
-x	Protect the RAM disk via HIMEM.
-z	Remove RAM disk but not the TCP/IP BOOT-PROM memory.
-?	Display the PXUTIL options.

Patching text files with individual BOOTP/DHCP Information

The information that the TCP/IP BOOT-PROM obtains from the BOOTP/DHCP Server can be processed with the BPUTIL program.

BPUTIL replaces placeholders with server provided, individual information from BOOTP/DHCP (standard and custom) options, e.g.

The text file with the following placeholders

```
This PC uses the IP address [#@yip#####] along
with the subnet mask #@smf#####.
```

looks like this after patching with bputil -a:

```
This PC uses the IP address [10.0.0.1 ] along
with the subnet mask 255.0.0.0 .
```

If the address components need to be separated by spaces instead of dots, bputil -

p can be used instead.

How does the replacement work ?

When replacing placeholders in files with individual information, BPUTIL must not change the file length. Therefore, the length of each placeholder must be specified with # characters. Be careful not to use too few # characters, or else the data will be truncated as in the following example.

The line

```
This PC uses the IP address [#@yip#] along with the
subnet mask #@smf#.
```

Becomes after replacement with bputil -a

```
This PC uses the IP address [10.0.0] along with the
subnet mask 255.0..
```

Using # signs within placeholders, make sure that enough space is reserved for the actual information to fit in. If the data is shorter than the placeholder, you may use an asterisk * to shift trailing spaces to the end of the line.

The line

```
This PC uses the IP address [#@yip*#####]
along with the subnet mask #@smf*#####.
```

Becomes after replacement with bputil -a

```
This PC uses the IP address [10.1.0.1] along with the
subnet mask 255.0.0.0.
```

Transfer BOOTP/DHCP options to environment variables

Using BPUTIL -s, information from BOOTP/DHCP options can easily be transferred to environment variables.

BPUTIL -s shows all BOOTP/DHCP options in a format that can be directly used to set environment variables, e.g. bputil -s yip shows the client's IP address in the following format:

```
set yip=10.0.0.1
```

By redirecting the output to a batch file and calling this batch file, the corresponding BOOTP/DHCP information can be made available as an environment variable:

```
bputil -s yip > m.bat
call m.bat
del m.bat
echo my IP address is %yip%
```

BMDRV / BMUTIL32

BMDRV is a Windows device driver that makes the BOOTP/DHCP information accessible from within these operating systems.

BMUTIL32 allows patching text and binary files as well as the Windows Registry with the BOOTP/DHCP reply information. BMUTIL32 is the Windows version of the DOS based PXUTIL/BPUTIL utilities.

BMUTIL32 Command Line Switches

-?	Show BMUTIL32 usage
-a	patch BOOTP/DHCP reply information into ASCII file
-b	patch BOOTP/DHCP reply information into binary file
-c	show BootManage bootstrap loader configuration information
-C	check if the BMDRV device driver is running
-d	as -r but shows debugging information and does not modify the Registry
-D	as -R but shows debugging information and does not modify the Registry
-e	disables access to the BMDRV device driver
-f	flushes the Windows NT/2000 in-memory Registry copy to disk
-install	install BMUTIL32 as Windows NT/2000 service
-p	patch BOOTP/DHCP reply information into ASCII file
-parms	show BMUTIL32 Windows NT/2000 service startup parameters
-r	insert information from ASCII file into the Registry
-R	insert patched (DHCP/BOOTP) information into the Registry
-remove	remove BMUTIL32 Windows NT/2000 service
-run	start the BMUTIL32 Windows NT/2000 service
-s	show all or selected BOOTP/DHCP reply information
-S	show only the BOOTP/DHCP reply information that is available
-stop	stop the BMUTIL32 Windows NT/2000 service

BMFDISK

The BMFDISK utility allows partitioning and formatting a hard disk, query and setting individual partition table information, and performing a number of other low level hard disk manipulations.

Command Line Switches

-?	display BMFDISK usage
-a	add a +/- value to part. ID (ERRORLEVEL: new part. ID)
-b	write master boot record
-c	check for existence (ERRORLEVEL: no = 0, yes = 1)
-d	clear partition boot sector
-g	display disk geometry
-G	same as -g but does not display error message if fails
-i	search for partition
-m	make a partition
-o	set partition ID
-p	display partition table
-P	display partition table using environment variable format
-q	quick format a partition (write boot sector and FATs/DIR)
-r	read from the disk and write to a file
-t	set partition to active
-w	read from a file and write to the disk
-z	write zero (sectors with content 0) to the disk

Option / Parameter Explanation

b#	a decimal number specifying an absolute block
blks	a decimal number specifying a number of blocks
drv	a hexadecimal drive number
part	a hard disk partition number (0, 1, 2 or 3)
id	a hexadecimal partition ID
-s b#	specifies the starting block number
-e b#	specifies the ending block number
-n blks	specifies the number of blocks
-l kbs	limit throughput to kbs kilobytes per second
-f	force immediate execution, don't delay to allow user abort
-v	display more information upon command execution

Setup Hard Disk

Create partitions

BMFDISK -m 80 0 n e1 8m -f

BMFDISK -m 80 1 y 06 2000m -f

BMFDISK -m 80 2 n 07 r -f

Delete partition

BMFDISK -m 80 2 N 0 c

Quick format partition

BMFDISK -q 80 1 fat16 -f

Write master boot record

BMFDISK -b 80 -f

Read Hard Disk information

Geometry

BMFDISK -g 80

Partition information

BMFDISK -p 80

Test partition ID

BMFDISK -c 80 0 e1

Example

```
bmfdisk -c 80 0 e1
if ERRORLEVEL 1 goto HD_FORMAT
bmfdisk -c 80 0 e2
if ERRORLEVEL 1 goto START_NETWORK
```

Modify Partition ID

Set partition ID

BMFDISK -o 80 0 e1-f

Increase/decrease partition ID

BMFDISK -a 80 0 1 -f

BMFDISK -a 80 0 -1 -f

BMPCSCAN

BMPCSCAN is a utility to assist in the automated driver installation by obtaining information about PCI and onboard PNP devices. It also allows looking for specific devices and provides the hardware information to unattended installation batch scripts.

Command Line Switches:

bmpcscan [pci|pnp] [-s] [-t type] [-f fname] [-v vlevel]

pci	Scan for PCI devices.
pnp	Scan for embedded Plug-and-Play devices.
-s	Use environment variable output format. If this option is not present, use report output format instead.
-t	type Only display information about devices of the given type
-f	fname Full pathname of database file (pcicode.dat / pnpcode.dat)
-v	vlevel Verbosity level (0 ... 2)

```
bmpcscan pci -v 2
-----
Bus Device BaseClass SubClass ProgIf VendID DevID SubsysID RevID
01      00      03      00      00    1002  474d  00041002   65
Company : ATI ( ATI TECHNOLOGIES INC )
Device  : ATI Technologies RAGE XL AGP 2X
Type    : Generic VGA compatible
```

```
bmpcscan pci -s -v 0
SET PCI_NET0=80861229
SET PCI_VGA0=1002474d
SET PCI_MMDAUD0=1073000c
SET PCI_BRIDGEHOST0=80867190
SET PCI_BRIDGEISA0=80867110
SET PCI_BRIDGEPCI0=80867191
SET PCI_BRIDGE0=80867113
```

```
bmpcscan pci -t NET -s -v 1
SET PCI_NET0v=8086
SET PCI_NET0d=1229
SET PCI_NET0s=30008086
SET PCI_NET0r=08
```



Integrating the BMA in custom applications

ActiveX Interface

The ActiveX interface has been developed to integrate the BMA in custom applications. This way, the BootManage Administrator can be controlled through DCOM from any authorized location within a network.

DCOM (Distributed Component Object Model) uses the Microsoft security concept. The security settings can be configured using the DCOMCNFG application.

Within the BootManage Administrator program directory, you will find a Visual Basic Script that explains all the available functions.

```
Dim objActiveXServer
Dim l

Set objActiveXServer =
CreateObject("BMA_Server.clsWriteData", "localhost")
l = objActiveXServer.ReinstallClient("00de2e870320")

MsgBox "Return value: " & l
Set objActiveXServer = Nothing
```

Interface Description of BMA_Server

Error codes:

10	INVALID_TFTPBOOT	TFTPBOOT entry missing in server
20	INVALID_MAC	Wrong MAC address
21	INVALID_NAME	Wrong client name
22	INVALID_CLIENT_PARAM	Wrong client parameter
25	INVALID_SERVERNAME	Wrong server name
40	INVALID_OS	Wrong operating system name
41	INVALID_OS_PARAM	Wrong operating system parameter
50	INVALID_BMSTRUCT	Wrong boot image structure
150	INVALID_DATABASE	Error opening database
151	NO_DATA_FOUND	No data record found
100	ERROR_WRITE_FILE	Error writing to file
1000	ERROR_LICENSE	License error or no free licenses
1001	ERROR_DEFAULT_PARAM	Wrong database settings

Supported Functions

Show all Client MAC Addresses

Optionally, redirect this command to a different server.

Object: BMA_Server.clsGetData

Function: GetAllClients(retMAC\$, retClientNames\$, retNumberClients, Optional ServerName\$ = "", Optional MainGroup\$ = "")

RetMAC	Return all managed client PC's MAC addresses. Separator character is 0x00
retClientNames	Return all managed client PC's names. Separator character is 0x00
retNumberClients	Return the number of all managed client PCs
ServerName	Optional: only return information about PCs that are assigned to the specified server
MainGroup	Optional: only return information about PCs that are assigned to the specified group

Show all available operating systems of a server

Object: BMA_Server.clsGetData

Function: GetOSNames(RetBuffer\$, RetNumberOfOS, ServerName)

RetBuffer	Return operating system name. Separator character is 0x00
RetNumberOfOS	Return number of operating systems
ServerName	Return the server's name

Show all server names

Object: BMA_Server.clsGetData

Function: GetServerNames(RetBuffer\$, RetNumberOfServer)

RetBuffer	Return all server names. Separator character is 0x00
RetNumberOfServer	Return number of servers

Show all Groups

Optionally, redirect this command to a different server.

Object: BMA_Server.clsGetData

Function: GetGroups(RetBuffer\$, RetNumberOfGroups, Optional ServerName\$ = "")

RetBuffer	Return group names. Separator character is 0x00
RetNumberOfGroups	Return number of groups
ServerName	Optional: only return information about groups that are assigned to the specified server

Show all client properties in an array

Object: BMA_Server.clsGetData

Function: GetClient(MAC\$, retData())

MAC	Client MAC address
retData()	Return client parameters as array
retTable()	Return table description as array

retData structure:

Mac

Name

FullName

LastInstall

LastStartup
 InstalledOS
 InstallServer
 ServerName
 MainGroup
 FirstPartitionSize
 FileSystem
 SerialNumber
 InsKey0
 InsKey1
 LastInsKey
 Comment
 ExtendOEMPartition
 Organization
 InstallOS
 BootImage
 Not used
 Not used
 Additional optional parameters that cannot be exactly specified

Show all server properties in an array

Object: BMA_Server.clsGetData

Function: GetServer(ServerName, retData())

ServerName	Server name
retData()	Return server parameters as array
retTable()	Return table description as array

retData structure:

ServerName
 NewClientSourceDir
 FullAccess
 RoAccess
 SupportedOS
 Remark
 RealServerName
 InstallUsername
 InstallPassword
 Domain
 Workgroup
 OSMainPath
 TftpDestination
 FullName
 Organization

Show all operating systems in an array

Object: BMA_Server.clsGetData

Function: GetOS(OSName, retData())

OSName	Operating system name
retData()	Return operating system parameters as array
retTable()	Return table description as array

retData Array structure

Name
 FullName
 Comment
 LocationPath
 SerialNumber
 FirstPartSize
 SupportedFileSystem
 ExtendOEMPartition

Version

PC Wake-On-LAN

Object: BMA_Server.clsNetwork
Function: SendWakeOnLan(MAC\$)

MAC	Client MAC address
-----	--------------------

PC Wake-On-LAN via lookup zone

Object: BMA_Server.clsNetwork
Function: SendWakeOnWan(MAC\$)

MAC	Client MAC address
-----	--------------------

Modify client operating system

Object: BMA_Server.clsWriteData
Function: ChangeClientOS(MAC\$, OSName, SerialNumber)

MAC	Client MAC address
OSName	New client operating system name
SerialNumber	Serial number for the new operating system

Reinstall client PC

Object: BMA_Server.clsWriteData
Function: ReinstallClient(MAC\$)

MAC	Client MAC address
-----	--------------------

Sample ActiveX call across the network

```
Private Function GetServer()  
    Dim objActiveXServer As Object  
    Dim RetBuffer As String  
    Dim NumberOfServer As Long  
    Dim ActiveXServer as String  
    Dim l As Long  
  
    ActiveXServer = "Thinkpad"  
    Set objActiveXServer =  
CreateObject("BMA_Server.clsGetData", ActiveXServer)  
    l = objActiveXServer.GetServerNames(RetBuffer$,  
NumberOfServer)  
    If l = 0 Then  
        MsgBox "Anzahl der Server:" & NumberOfServer  
& vbCrLf & RetBuffer$  
    Else  
        MsgBox "Fehler: " & l  
    End If  
    Set objActiveXServer = Nothing  
End Function
```



Webserver

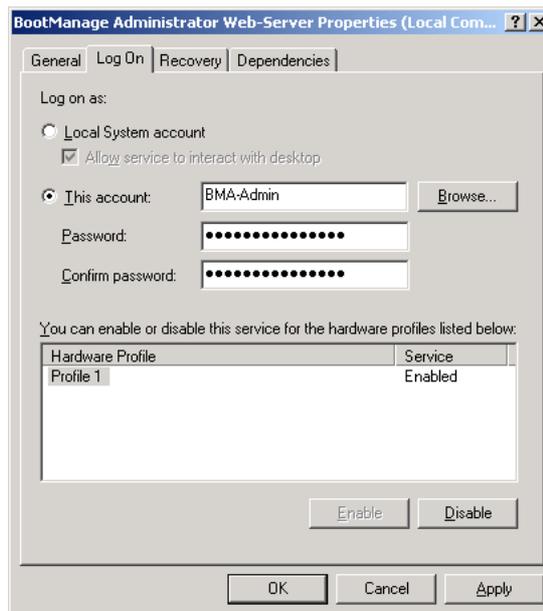
Overview of the BMA Webserver functions

The web server allows location independent administration of the BootManage Administrator. All you need in order to manage your client PCs with the BootManage Administrator is a standard web browser.

Starting the BMA Web Server

The web server is installed together with the BootManage Administrator and can be operated in two different modes:

- 1.) Application with status window
For testing purposes, or if you want to actively monitor server operation, it is useful to start the BMA web server as application. Click on „Start,Program Files,bootix Technology GmbH, BootManage Administrator, BMA Webserver“
- 2.) Service
To install the BMA web server as service, click on „Start,Program Files,bootix Technology GmbH, BootManage Administrator, Install BMA Webserver as Service“. When operating as service, make sure that the BMA web server runs in the context of a user who has the necessary access permissions to the shares that the BMA requires.



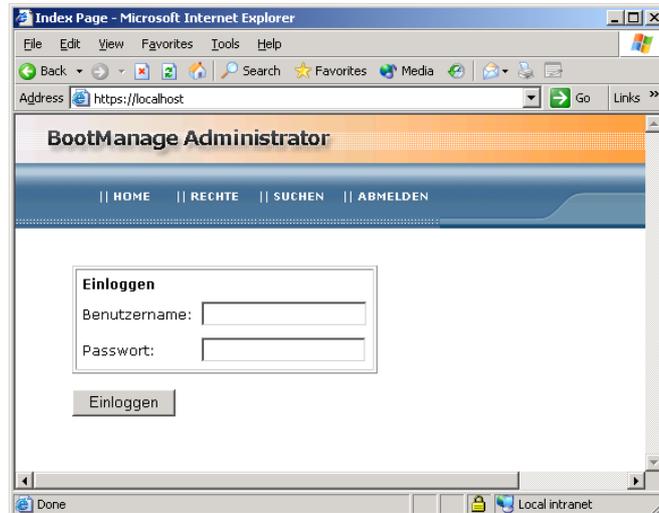
Client Management with the BMA Web Server

After the BMA web server is started, it can be accessed from an arbitrary location with a standard web browser. The connection is always encrypted (SSL), so that security is granted even when managing over the internet. To allow only authorized personnel access to the BMA web server, a login is required.

To access the BMA web server from your browser, simply enter the URL https://name_of_bma_server.

HTTPS means that an SSL (secure socket layer) encrypted secure connection is established between browser and BMA web server. Due to the encryption, certificates need to be exchanged on first connection. These certificates have to be accepted only once. After that, the login screen is displayed in your browser.

Log in using your BMA user name and password, and you can manage the BMA functions through your web browser.



Configuring the BMA Web Server

Upon start-up, the BootManage Administrator reads its configuration from the text based configuration file %ProgramFiles%\bootix\httpd.conf. You may edit this file with a text editor in order to adapt the web server's behaviour to suit your needs.

The default ports that the web server operates on are port 80 for standard HTTP connections and port 443 for SSL encrypted HTTPS connections. If another web server runs on the same machine as the BootManage Administrator web server, you may want to change these port assignments in order to avoid conflicts.

In this case, simply change the settings "HttpPort=80" and "SSLPort=443" within the [HTTPServer] section to different port numbers and restart the web server. Of course, when using non-standard port numbers, you must specify these port numbers in your web browser when connecting to the BootManage Administrator web server.

If you want to disable SSL connections, just set the parameter "EnableSSL" to the value "False". The default value is "True".

When establishing an SSL encrypted HTTPS connection with the BootManage Administrator web server for the first time, it will present a certificate to your browser and ask you if you are willing to accept it. After accepting the certificate once, all further SSL connections to the web server from your browser will not show this dialog again.

BMA Webserver Roll-Out

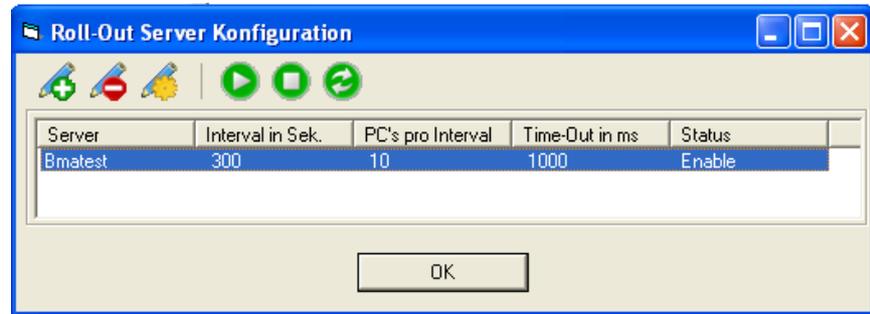
The Roll-Out service configuration is performed as usual via the import of a CSV file (Tools > Automated Roll-Out > Configure). The Roll-Out service that is integrated in the BMA console is no longer automatically started after CSV file import.

The new menu entry (Tools > Automated Roll-Out > Server Configuration) provides configuration of the new BMA Web-Server Roll-Out function – separately for each installation server that is registered in the BMA.

Note that there is only one central roll-out service that is implemented as an ActiveX component and integrated in the BMA web server. The roll-out service

“sits” within the web server. The BMA-administered installation servers are not active components, but Windows shares which are serviced in turn by the BMA roll-out service. For each BMA installation server can be defined individually if and how this server is serviced by the BMA roll-out service. The administration of this configuration is performed via the BMA console (Tools > Automated Roll-Out > Server Configuration), but the roll-out service itself can now be run within the BMA web server.

Conclusion: The BMA console need no longer be running during the roll-out.



The first three symbols refer to the BMA installation server:

Pen with green plus sign: Create roll-out configuration for BMA install server

Pen with red minus sign: Delete roll-out configuration for BMA install server

Pen with yellow gearwheel: Edit roll-out configuration of BMA install server

The second three symbols refer to the roll-out service itself:

Green play symbol: Release roll-out service

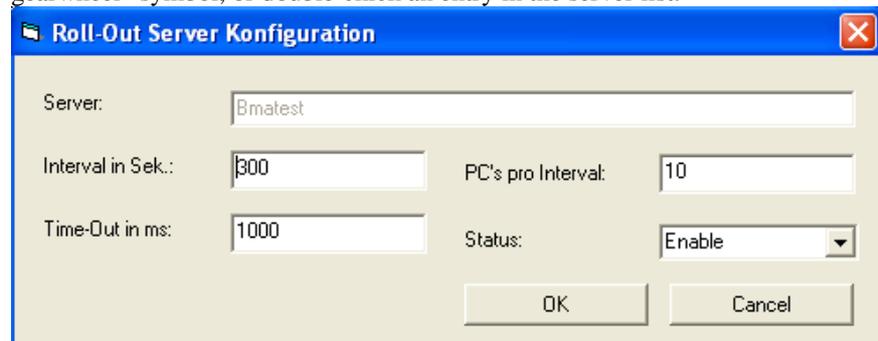
Green stop symbol: Lock roll-out service

Green reload symbol: Refresh display

Note that, with the green play and stop symbols, the roll-out service is only globally released or locked, but no actual service is started or stopped. The actual roll-out service itself must be started or stopped from within the web server.

Conclusion: A roll-out operation is only active if the roll-out configuration is released *AND* the roll-out service is started within the web server.

To display the server configuration dialog, click on the „pen with yellow gearwheel“ symbol, or double-click an entry in the server list.

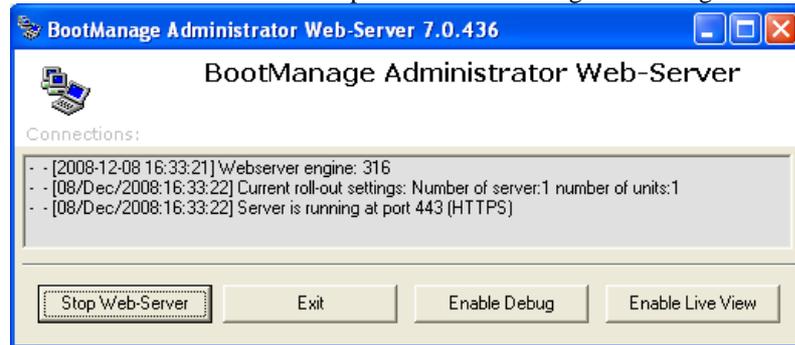


Here, the roll-out configuration is defined for each BMA server individually, i.e.:

- whether this server is serviced by the roll-out service at all (Status enable/disable)
- in what time intervals a check for new clients is performed (Interval in sec.)
- how many new clients are serviced at the same time (PCs per Interval)
- after what timeout value a server is considered as unreachable (Time-Out in ms)

When the roll-out server configuration is finished, the BMA console can be closed.

The first start of the BMA web server should be performed as application, so that one can observe the actions and possible error messages in the log window.



During the start, the above messages should be displayed.

When everything works, the BMA web server can be installed as Windows service, and then started.

Start your web browser, connect to the BMA web server, and logon to it. Then, in the menu bar, click „ROLL-OUT“.

In the „Service“ tab, the roll-out service can be stopped and started.

The „Server configuration“ tab displays the roll-out configuration of the BMA installation servers.

The „Clients“ tab gives an overview of the roll-out status for the various PCs, i.e. which clients are scheduled for a roll-out, and whether the roll-out was already performed for a certain client.

Finally, the „Logfile“ tab displays the BMA installation server's roll-out log file.



Modifying the Web View

The access rights to server, groups and PCs are configured within the BootManage Administrator. One can control what is visible to the user, and what is hidden. In some cases, it may be desirable to restrict users, e.g. to hide a serial number or to prevent a user from changing a PC's name.

The BMA web server provides an efficient means to adapt the web view, thus making it possible to provide different web views for different users. The BMA web server generates the web pages from control files that contain HTML code, which simplifies the creation of individual web pages. The desired client information is masked by placeholders that the BMA web server replaces with the corresponding information whenever the page is called up.

The control files are located in the doc directory, directly below the BMA web server program directory.

Three different methods of individualization exist:

- Dependent on the operating system
- Dependent on the user
- Dependent on both the operating system and the user

For all three methods, individualized copies of the control files must be placed in subdirectories of the central doc directory.

Dependent on the operating system

To create an individualized page of a certain operating system for all users, first create a subdirectory named „os“ in the central „doc“ directory“. Within the „os“ directory, create a subdirectory with the name of the desired operating system. For this, use the name as it is registered in the OS.INI file (e.g. w2k_pro, xp_pro, w3k_s, etc).

Example:

- Within the “doc” directory, create a subdirectory named „os“.
- Within the “os” directory, create a subdirectory named „xp_pro“.
- Copy the file „client.bma“ to the “xp_pro” subdirectory.
- Open the just-copied „client.bma“ file with a text editor.
- Look for the text <BODY>
- Below the <BODY> line, insert a new line with the following text:
Individualized for operating system: xp_pro
.
- Save the file and point your web browser to the BMA web server.
- Logon as „admin“ and display the client properties of a PC that was installed using the “xp_pro” operating system.
- In the first line, you will see the text that you entered.

Dependent on the user

To provide a certain user with an individualized page, first create a subdirectory named „user“ in the central „doc“ directory“. Within the „user“ directory, create a subdirectory with the user's logon name.

Example:

- Within the “doc” directory, create a subdirectory named „admin“.
- Copy the file „client.bma“ to the “admin” subdirectory.
- Open the just-copied „client.bma“ file with a text editor.
- Look for the text <BODY>
- Below the <BODY> line, insert a new line with the following text:
Individualized for user: admin
.
- Save the file and point your web browser to the BMA web server.
- Logon as „admin“ and display the client properties of a PC.
- In the first line, you will see the text that you entered.

Dependent on both the operating system and the user

To provide a certain user with an individualized page of a certain operating system, first create a subdirectory named „os“ in the central „doc“ directory“. Within the „os“ directory, create a subdirectory with the name of the desired operating system. For this, use the name as it is registered in the OS.INI file (e.g. w2k_pro, xp_pro, w3k_s, etc). Within this directory, create a subdirectory with the user's logon name.

Example:

- Within the “doc” directory, create a subdirectory named „os“.
- Within the “os” directory, create a subdirectory named „w2k_pro“.
- Within the “w2k_pro” directory, create a subdirectory named „admin“.
- Copy the file „client.bma“ to the “admin” subdirectory.
- Open the just-copied „client.bma“ file with a text editor.
- Look for the text <BODY>
- Below the <BODY> line, insert a new line with the following text:
Individualized for operating system: w2k_pro and user: admin
.
- Save the file and point your web browser to the BMA web server.
- Logon as „admin“ and display the client properties of a PC that was installed using the w2k_pro operating system.
- In the first line, you will see the text that you entered.

Precedence rules for individualized files

When processing control files, the BootManage Administrator web server uses the following precedence rules.

First, it looks for the file in the “doc\os\Next, it looks for the file in the “doc\os\Then, it looks for the file in the “doc\user\Finally, it looks for the file in the “doc” directory

The first found file is taken, so the above order reflects the precedence rules.

Placeholders within control files

The placeholders for the actual data have been inserted into the control files in the form <!--#name_of_placeholder --> so that HTML editors recognize them as comments.

There are three types of placeholders:

- 1) Normal: <!--#name_of_placeholder -->
This placeholder shows you, depending on your access permissions, only the text, or allows editing the data field.
- 2) Read only: <!--# name_of_placeholder TYPE=READONLY -->
This placeholder shows only the text of the data field.
- 3) Hidden: <!--# name_of_placeholder TYPE=HIDE -->
This placeholder allows to hide information. The data field is only visible in the source code of the web page.



Boot Image Management

In release 7.4, the BootManage Administrator provides a completely redesigned and dramatically improved Boot Image Management module that allows using the following boot image types side-by-side:

- DOS/Win98
- BartPE 32-Bit x86
- Windows PE 2005 32-Bit x86
- Windows PE 2005 64-Bit x64
- Windows PE 2.0 32-Bit x86
- Windows PE 2.0 64-Bit x64
- Windows PE 3.0 32-Bit x86
- Windows PE 3.0 64-Bit x64

One can define multiple boot images of the same type, and assign them to BMA servers, groups, clients and operating systems as needed.

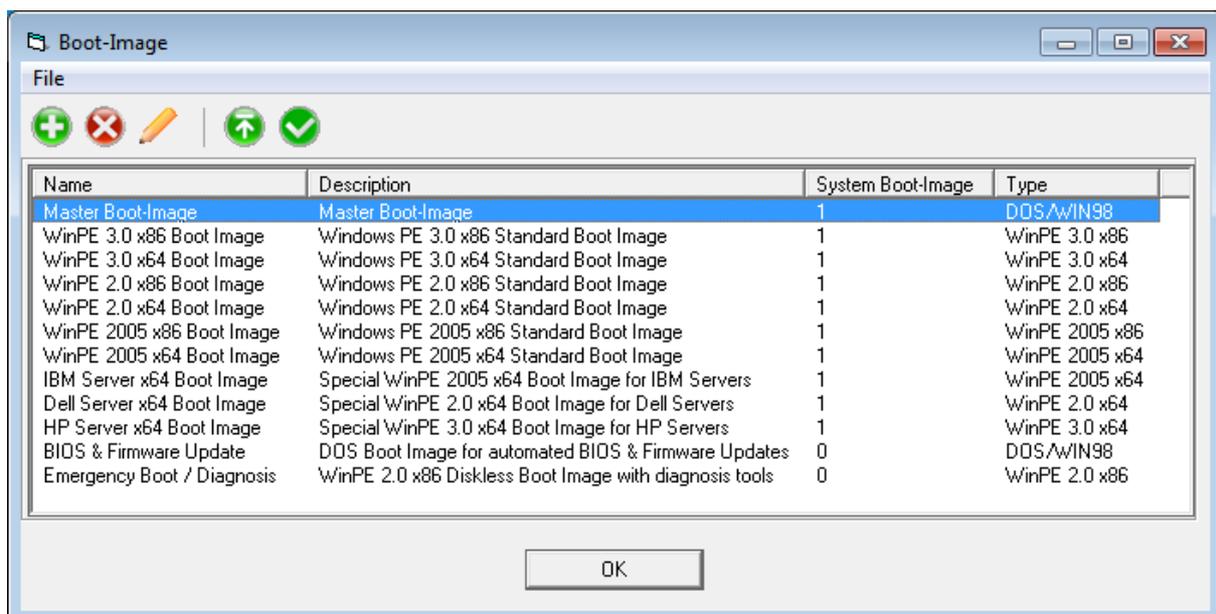
Also, BootManage Administrator 7.4 comes with graphical WinPE Boot Image Creator applications, so it is no longer necessary to create WinPE boot images manually from batch scripts (although you can continue to use this method). The WinPE Boot Image Creator applications produce WinPE boot images with a couple of mouse clicks, and automatically integrate the bootix tools for WinPE as well as third-party drivers. For detailed information, please consult the documentation that comes with the WinPE Boot Image Creator applications.

Overview

There are several reasons why you may have to use multiple boot images of different types, for example:

- Starting with Windows Vista, Microsoft does no longer support unattended Windows installations over DOS/Win98.
- Windows x64 operating systems require WinPE x64 for unattended installation, and cannot be installed using WinPE x86 (and vice versa).
- Some machine types do not support DOS/Win98 at all
- Unattended BIOS & Firmware Updates, emergency boot and hardware diagnosis boot images require special drivers and integrated tools

Once a boot image is defined, imported and distributed, it can be easily assigned to BMA servers, groups, clients and operating systems. In this section, we will explain all these steps in detail. Here is a first look at the new Boot Image Management main dialog, already populated with some sample boot images:



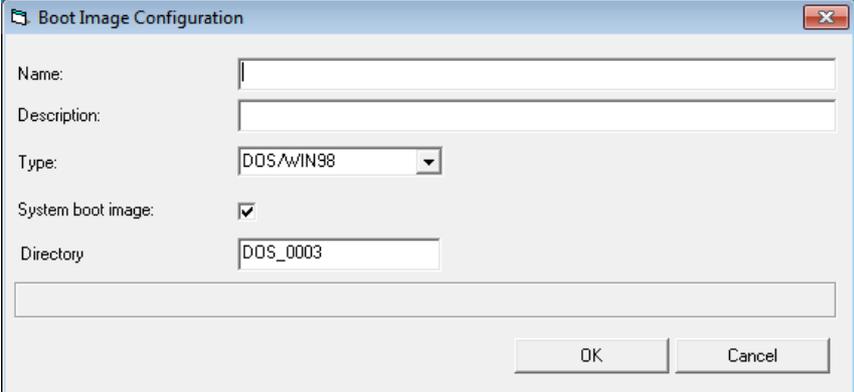
From the File menu, the Windows PE Boot Image Creator applications (and their help files) can be launched.

Using the buttons below the menu bar, the following tasks can be performed:

- Define a new boot image entry
- Delete an existing boot image entry
- Edit an existing boot image entry
- Distribute boot images to BMA servers
- Check boot image consistency on BMA servers

Define New Boot Image Entry

When you click the “Add Boot Image” button (white plus sign in green circle), the following dialog is displayed:



First, type a name and description for the new boot image. This is used to later identify and assign the boot image to BMA servers, groups, clients and operating systems.

The “System boot image” checkbox indicates whether this boot image can be used for unattended operating system installations in the context of the BootManage Administrator. Such boot images need to contain the bootix tools and scripts in order to communicate with the BMA server and console, so that proper client import, hardware detection, unattended installation control and status reporting is possible. The already present default DOS/Win98 boot image is a system boot image. Also, all Windows PE boot images that are created with the Windows PE Boot Image Creator tools are capable of being system boot images.

The “Directory” specifies the folder in which the boot image files are internally managed. It is recommended to leave this setting as proposed.

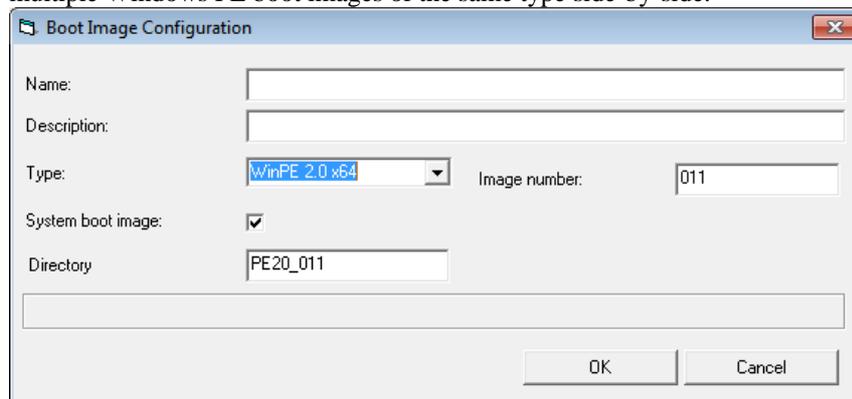
Finally, select the boot image type. The drop-down box lets you choose between DOS/Win98 and multiple WinPE boot image types. Note that BartPE is not separately listed. For BartPE boot images, use the “WinPE 2005 x86” type. Depending on the selection, the contents of the dialog box changes to let you enter additional type specific settings.

DOS/Win98 Specific Settings

For DOS/Win98 boot images, there are no additional settings, and the dialog box looks as displayed above. It is not recommended to change the “Directory” entry, but if you feel the need to do so, please note that only 8 characters are allowed for DOS/Win98 boot images.

Windows PE Specific Settings

Regardless of their type, all Windows PE boot images require a so-called “Image number” as additional setting. This is a technical requirement for allowing multiple Windows PE boot images of the same type side-by-side.

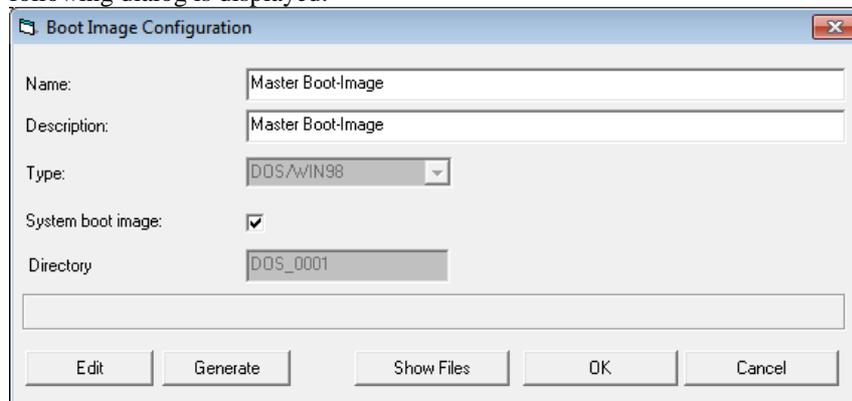


It is not recommended to change the image number, but if you feel the need to do so, please note that it must always consist of exactly three characters. Since each Windows PE boot image must have a unique image number, it is not possible to assign the same image number to multiple Windows PE boot images.

It is also not recommended to change the “Directory” entry, but if you feel the need to do so, please note that only alphanumeric and some additional special characters are allowed here. As opposed to DOS/Win98 boot image types, more than 8 characters are allowed.

Edit & Generate DOS/Win98 Boot Image

When you define a new DOS/Win98 Boot Image or edit an existing one, the following dialog is displayed:



The name and description texts, and also the system boot image status can be changed, but not the type and directory settings.

The buttons at the bottom of this dialog box allow editing, generating and displaying the files of a DOS/Win98 boot image.

Edit DOS/Win98 Boot Image

When you click the Edit button, an explorer window opens and displays the contents of the corresponding DOS/Win98 boot image’s source folder. Note that this is not the boot image itself, but the reference folder from which the actual boot image is generated. Add, change or delete files here to satisfy your special needs, but make sure that you know what you are doing. Modifying boot images (especially system boot images) has significant effect on the way that unattended operating system installations and other projects work. When you are done, close the Windows explorer window.

Generate DOS/Win98 Boot Image

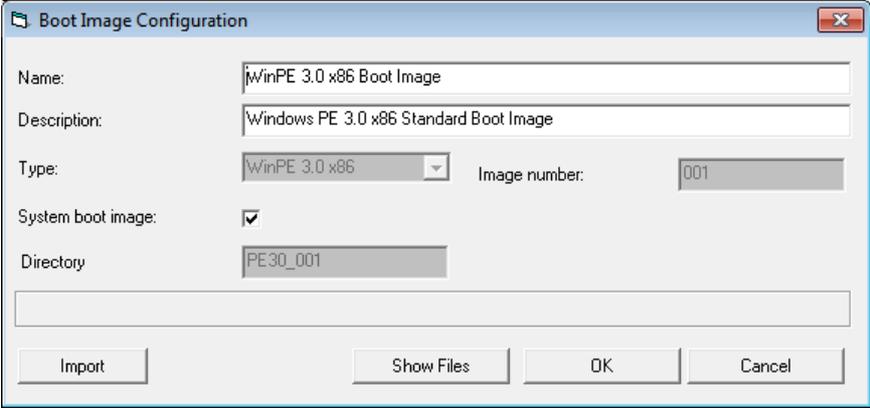
Clicking the Generate button starts the DOS/Win98 boot image generation process, i.e. the contents of the reference folder are converted to an actual boot image file. Note that the boot image is only generated, but not automatically distributed to the BMA TFTP servers. This must be done separately.

Display DOS/Win98 Boot Image Files

When you click the “Show Files” button, an explorer window opens and displays the contents of the corresponding DOS/Win98 boot image’s destination folder. This is where the actual generated boot image files are located. From there, the boot images will be distributed to the BMA TFTP servers.

Import Windows PE Boot Images

Windows PE boot images are much more complex than DOS/Win98 boot images, and cannot be edited directly in the BMA Boot Image Configuration dialog. Instead, there are separate Windows PE Boot Image Creator tools available that can be launched from the File menu. These tools create all the files that make up a Windows PE boot image, and place it in an “image” folder. The BMA Boot Image Configuration dialog is then used to import these files into the BMA. Therefore, when handling WinPE boot images, the dialog shows an “Import” button instead of the DOS/Win98 specific “Edit” and “Generate” buttons.



When you click the “Import” button, a file selector box opens and asks you to specify the folder that contains your WinPE Boot Image files. If you used one of the Windows PE Boot Image Creator tools for creating the WinPE boot image (which is highly recommended), this folder will be either “image\x86” or “image\x64”, depending on the WinPE boot image’s architecture. After clicking OK, all WinPE boot image files will be imported into the BMA. Note that the WinPE boot image is only imported, but not automatically distributed to the BMA TFTP servers. This must be done separately.

Delete Existing Boot Image Entry

If you want to delete an existing boot image entry, first select the corresponding entry and then click the delete button (white ‘x’ in red circle). In order to prevent accidental deletion, a safety dialog will request you to confirm the deletion.

System Boot Images cannot be deleted. In order to delete a system boot image, you must previously turn it into a standard boot image (by removing its system boot image status).

Edit Existing Boot Image Entry

Click on the pencil button to edit an existing boot image entry.

Distribute Boot Images To TFTP Servers

Once a boot image is created or imported, it is available in the BMA and ready for distribution to the TFTP servers. This distribution process is not performed automatically, but must be initiated manually.

Select one or more boot images to be distributed to a certain BMA TFTP server, and then click the Distribute button (white up arrow in green circle). In the dialog that opens, select the target BMA TFTP servers and click OK. The selected boot images are now distributed to all specified BMA TFTP servers. After that, these BMA TFTP servers are capable of providing these boot images to requesting PXE clients.

Check Boot Image Consistency On TFTP Servers

You may want to determine whether specific boot images are present or up to date on BMA TFTP servers. In this case, first select the boot images and then click the “Check consistency” button (white checkmark in green circle). In the dialog that opens, select the target BMA TFTP servers and click OK. The presence and consistency of the selected boot images is now checked on the target servers, and a report is generated. From this report, you can see whether you need to take further action or not.

Assign Boot Images

Once that boot images are created or imported and distributed to BMA TFTP servers, they can be assigned to BMA servers, groups, clients or operating systems in the BMA console.

Boot Image assignment is performed via the BMA special variable “T149” respectively „Boot Image“, so you can always use this variable to define boot image assignment at each object level.

However, for some objects, the BMA console offers a more comfortable way to assign boot images:

Boot Image Assignment for BMA Servers

In the server properties dialog, click the “Advanced” tab. There, you can select the server’s default boot image from a drop-down list.

Boot Image Assignment for Clients

In the client properties dialog, click the “Boot-Image” tab. There, you can select the client’s explicitly assigned boot image from a drop-down list, or select “Use default system Boot-Image” instead.

Boot Image Assignment for Groups & Operating Systems

For groups and operating systems, there is no special setting for boot image selection. Here, you must assign the system variable „Boot Image“ to the selected group or operating system, and set its value to the desired boot image.

Globally Assigned Boot Image

If you want to assign a special boot image as global default for the entire BMA system, proceed in the same way like for groups and operating systems, i.e. use the global system variable „Boot Image“, and set its value to the desired boot image.



BartPE Boot Images

As an alternative to Windows PE 2005, the BootManage Administrator also supports BartPE boot images that have been created with PE Builder v3. The BootManage Administrator distribution comes with a PE Builder v3 plugin.

Please note that PE Builder itself is not part of the BootManage Administrator distribution package. In order to obtain PE Builder, please visit the web site <http://www.nu2.nu/pebuilder>. There, you will also find product and licensing information about PE Builder and BartPE. As of this writing, the current version of PE Builder is 3.1.10a.

We assume that you are already familiar with PE Builder, plugins, and BartPE ISO images. At this time, the BartPE boot image must be created manually, along with a few other tasks. The following step-by-step instructions will describe what needs to be done.

Prerequisites

Please make sure that you have the following products available:

- PE Builder v3
- A Windows Server 2003 with integrated Service Pack 1 product CD
- The BootManage Administrator plugin for PE Builder

Also, you will need a Windows (preferably Windows XP) PC on which to build the BartPE boot image.

Step 1) Install BootManage Administrator

By following the instructions in the BootManage Administrator installation manual, install the BootManage Administrator as usual. After that, you should have a standard BMA environment (DHCP Server, TFTP Server, Install Server, BMA Console) along with the well-known DOS/Win98 based default master boot image.

Step 2) Extend DHCP Server Configuration

On the DHCP server, set the standard option 066 to the IP address of the machine on which the TFTP service is running. On Unix/Linux based DHCP servers, use the 'next-server' option instead. This is necessary even if the TFTP service is running on the same machine as the DHCP service!

If option 066 is already defined (because your TFTP service is running on a different machine), you won't have to change anything.

Note: Instead of specifying the IP address, you may also use a 'resolvable host name'. In any case, make sure that the given name can be resolved into an IP address by the DHCP server!

Step 3) Install PE Builder

Install the PE Builder application. We assume that the target directory is c:\pebuilder.

Step 4) Add bootix BootManage Administrator plugin

Add the bootix BootManage Administrator plugin by copying the directory "bootix bma" from the product CD's "bartpe_files\plugin" directory to the c:\pebuilder\plugin directory.

Step 5) Modify PENETCFG plugin configuration

In order to automate network initialization and synchronize plugin execution order, we must make two adjustments to the PENETCFG plugin. The goal of these modifications is to make sure that the network interface is up and running when the BootManage Administrator script is executed as the last action of the

BartPE startup process. For background information, please refer to the corresponding information at the end of this chapter.

Step 5a) Automate network initialization

Open the file 'c:\pebuilder\plugin\penetcfg\penetcfg.ini' with a text editor and modify the settings in the [General] section as follows:

```
AutoStartNet=Yes
PromptForProfile=No
ShowGUI=No
```

Step 5b) Synchronize plugin execution order

In the c:\pebuilder\plugin\penetcfg directory, rename the file 'autorun0penetcfg.cmd' to 'autorun8penetcfg.cmd'.

Then, open the file 'c:\pebuilder\plugin\penetcfg\penetcfg.inf with a text editor and change the single occurrence of 'autorun0penetcfg.cmd' to 'autorun8penetcfg.cmd'.

Step 6) Add 3rd party drivers (optional)

If your target PC hardware requires the integration of one or more third party drivers (e.g. network or mass storage drivers) in the BartPE boot image, please refer to “Adding 3rd party drivers to BartPE” at the end of this chapter.

Step 7) Create BartPE boot image

Now, start PE Builder. Insert a “Windows Server 2003 with integrated Service Pack 1” product CD and select this as the Windows Installation Source. Make sure that the BootManager Administrator plugin is activated in the plugin list. Activate the creation of an ISO image and name this image “pebuilder.iso”. Then, create the image.

Step 8) Copy Files To TFTPBOOT Directory

Copy the just-created BartPE ISO image (pebuilder.iso) to the TFTPBOOT directory of your TFTP Server. Also, copy all files from the “bartpe_tftpboot” directory of the BootManager Administrator product CD to the TFTPBOOT directory of your TFTP Server. These files are:

```
. ntdetect.com
. ntldr
. winnt.sif
```

Step 9) Activate BartPE Boot Image

Activate the BartPE boot image for specific clients through the new “Use Windows PE as master boot image” checkbox in the client’s boot image property.

You may also want to instruct an entire installation server to use the BartPE boot image as default for all new clients. In this case, use the new “Use Windows PE as master boot image” checkbox in the server’s advanced property.

Automating Network Initialization

By default, the PENETCFG plugin requests interaction with the user when starting the network. In order to completely automate the BartPE boot image, we must change the PENETCFG settings so that the network initialization is performed completely automated.

Synchronizing Plugin Execution Order

The bootix BootManage Administrator plugin contains a script that must be executed as the very last action of the BartPE boot process, especially after the network has been initialized. For this, the “autorun” mechanism is used that is implemented by the “Startup Group / Autorun” plugin.

The BootManage Administrator script is run with autorun level ‘9’. However, the PENETCFG plugin which initializes the network runs asynchronously with autorun level ‘0’.

In order to make sure that the network initialization is completed before the BootManage Administrator script is executed, we must change the PENETCFG plugin’s autorun level to ‘8’.

Adding 3rd Party Drivers to BartPE

At this time, PE Builder supports to add network and mass storage drivers to a BartPE image.

Add 3rd party network driver

If the network adapter of your target PCs is not supported “out of the box” by BartPE (respectively by the operating system you use to build the BartPE image), you may need to add a third-party network driver by copying the driver files to the “driver\net” subdirectory. For details, please consult the PE Builder documentation.

Add 3rd party mass storage driver

If the mass storage (SCSI / SATA / SAS / RAID) adapter of your target PC is not supported “out of the box” by BartPE, you may need to add a third-party mass storage driver by copying the driver files to the “drivers\scsiadapter” subdirectory. For details, please consult the PE Builder documentation.



Client Prestaging

Overview

Starting with release 7.0, the BootManage Administrator provides a “client prestaging” function. Using a mechanism similar to the list based rollout, clients can be registered in the BootManage Administrator before they are actually installed. Using the prestaging function, a client appears in the BootManage Administrator console as if it were already installed using the BMA. When the client actually starts, it finds its configuration and performs the actual installation without requiring interaction with the BMA console. This is similar to the reinstallation of an already known client.

If you want to add clients via prestaging, you must provide a text based CSV file that contains the client properties, one line per client. This file has basically the same syntax as a list based rollout control file.

From the “Tools” menu, select “Prestaging”. You will be asked for the name of a prestaging control file. Use the “Browse” button to locate your file.

With the “Show” button, you can parse and display the prestaging file’s contents, and check its integrity. With the “OK” button, the prestaging file is accepted, and its contents are used to populate the BootManage Administrator with new “prestaged” clients.

Note that prestaged clients occupy one license each as soon as the prestaging information is imported in the BootManage Administrator.

Prestaged clients are displayed in the BootManage Administrator with a special icon. As soon as the prestaged client is actually provisioned, its prestage icon will change to a standard icon.

Prestaging sample files can be found on the BootManage Administrator product CD in the “samples\prestage” directory. You may want to use these files as templates and modify them to suit your needs.



SQL Server Support

Starting with release 7.0, the BootManage Administrator supports to store its database on an SQL Server. Prior to this release, the BootManage Administrator database was always provided as a Jet/Access MDB file.

When installing the BootManage Administrator, it will initially still use the Microsoft Jet/Access file based format, and store its database in the MASTER.MDB file which is located in the BootManage Administrator program directory (normally this is %ProgramFiles%\bootix).

Database Migration

If you want to transfer the BootManage Administrator database to an SQL Server, select “Options” from the “Tools” menu, and then click on the “Database” tab. The current database format and its properties are displayed in the dialog window. The only property of a Jet/Access database is its file name.

In order to transfer the BootManage Administrator database to an SQL Server, click the “Change Database Type...” button to start the Database Migration Wizard and select “Microsoft SQL Server” from the drop-down list.

SQL Server Properties

A Microsoft SQL Server database has four properties:

Server name

The SQL Server’s host name.

Database

The name of the database on the SQL Server. If this database does not already exist, it will be created as part of the migration process.

User Name

The name of a user who has the permissions on the SQL Server to create / administer /access the database. An account for this user must be present on the SQL Server.

Password

The above user’s password.

The Migration Process

After you have entered all the required information, click the “Next” button. The BootManage Administrator Database Migration Wizard performs a number of tests and then starts the actual database migration process while displaying a progress bar. After the migration is completed, the database tab displays the new properties.

Basically, you can perform a database migration at any time, but you should avoid concurrent access to the database during migration. For example, do not migrate the database when a roll out is in progress, or when other administrators work with the BootManage Administrator (e.g. via another BMA console, the BMA Web Server or the ActiveX interface). Especially, make sure that the BootManage Administrator web server is stopped during the migration.

You can also convert the database back from SQL Server to Jet/Access, from one SQL Server to another one, and also from one Jet/Access file to another at any time.

In case you want to observe the SQL communication between the BootManage Administrator and the SQL Server, either during a database migration or during normal operation after the migration, you may want to use the “MS SQL

Profiler” tool that comes with the Microsoft SQL Server. Using the Profiler, you can see the BootManage Administrator’s SQL queries and also the SQL Server’s replies to these queries.



Generating Reports

Starting with release 7.0, the BootManage Administrator contains a builtin report generator. Using predefined SQL queries, one can derive information from the BootManage Administrator database. This information can either be displayed as a table on the screen, or saved as a CSV, HTML, or XML formatted file.

The Report Generator Dialog

To open the report generator, select “Report Generator” from the “Tools” menu. From the drop-down list, select one of the predefined reports. If you want to display the report result as a table on the screen, click the “Show” button. In the report result window, you may sort the displayed lines by clicking on the table caption fields.

If you want to save the report result as a file instead, click the “Save as” button. In the file select box, enter a filename and select the file type which defines the format in which the report is to be saved.

Report File Formats

The following file types / formats are available:

CSV

A plain text file. Each entry takes up one line of text, fields are separated by a semicolon ‘;’. The first line contains the field names.

HTML

A web page. This file can be directly opened in a web browser and shows the entries in an HTML table.

XML

An XML file.

SQL Queries

Each report has a name and an associated SQL query string. Selecting a name from the drop down list also displays the associated SQL query string. It is also possible to use parameters in reports. When parameters are used, an additional drop down list for the parameter value appears below the SQL query window. In this case, select the parameter’s value before generating the report.

The SQL query string can be edited before clicking the “Show” or “Save as” buttons in order to generate the report. Note that modifications to the SQL query string are only temporary. When you select another report from the drop down list or close the report generator dialog window, your modifications are lost.

It is possible to add and remove reports, and also change the SQL query strings of existing reports. If you want to do this, open the file “report.ini” with a text editor. This file can be found in the BootManage Administrator program directory (normally this is %ProgramFiles%\bootix). In the [SQL Report] section, each report is listed on a separate line in the form <report name> = <SQL statement>. You may add, remove or modify lines to suit your needs, but be sure that you know what you are doing. In any case, it is recommended to create a backup copy of the report.ini file before making any modifications.

Note that the builtin reports only read information from the BMA database, they never write to it and so neither add nor alter information in the BMA database.

Templates for HTML Output

For the generation of web pages (HTML files), header and trailer files are used. The [Html Report] section contains references to these files. If you want to alter

the look of reports that are saved as web pages, you may modify the header and trailer files to suit your needs. The default header file is “report_h.dat”, and the default trailer file is “report_t.dat”.

Generating Rollout & Prestage template files

Tip: The report generator can be used to easily create control files for the list based rollout and prestaging functions. You may save a client report as CSV file and use this as a template for list based rollout and prestaging control files.



Troubleshooting

PXE-E32: TFTP open timeout.

The client does not find the TFTP server

```
Network boot from AMD Am79C970A
Copyright (C) 2003 VMware, Inc.
Copyright (C) 1997-2000 Intel Corporation

CLIENT MAC ADDR: 00 0C 29 F7 66 BD  GUID: 564D7DD6-5674-3DD0-6D12-58AA4DF766BD
CLIENT IP: 10.1.2.7  MASK: 255.255.0.0  DHCP IP: 10.1.2.1
PXE-E32: TFTP open timeout
TFTP.._
```

Possible cause:

The TFTP service on the server is not running.

If the TFTP and DHCP services are running on different machines, option 66 („Boot Server Host Name“ or „NextServer“) must be configured on the DHCP server.

PXE-T01: File not Found

The specified boot file does not exist.

```
Network boot from AMD Am79C970A
Copyright (C) 2003 VMware, Inc.
Copyright (C) 1997-2000 Intel Corporation

CLIENT MAC ADDR: 00 0C 29 F7 66 BD  GUID: 564D7DD6-5674-3DD0-6D12-58AA4DF766BD
CLIENT IP: 10.1.2.7  MASK: 255.255.0.0  DHCP IP: 10.1.2.1
TFTP.
PXE-T01: File not found
PXE-E3B: TFTP Error - File Not found
PXE-M0F: Exiting Intel PXE ROM.
Operating System not found
```

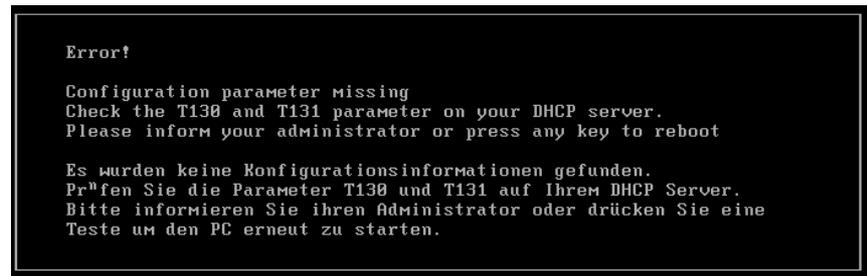
Possible cause:

During installation of the BootManage Administrator, the wrong TFTP directory was specified. From within the administration console, open the server properties and click on the button.

Use the following steps to verify and configure the TFTP directory.

- From the modify menu, select properties, server.
- In the server properties window, select the server and click the directories register.
- Verify and modify the exported TFTP directory, and click OK.
- From the tools menu, select manage boot image.
- In the boot image window, highlight manage master boot image, and click generate.

Configuration Parameter missing



Possible cause:

An error occurred while generating the OPT files.

The following steps allow regenerating the server OPT file:

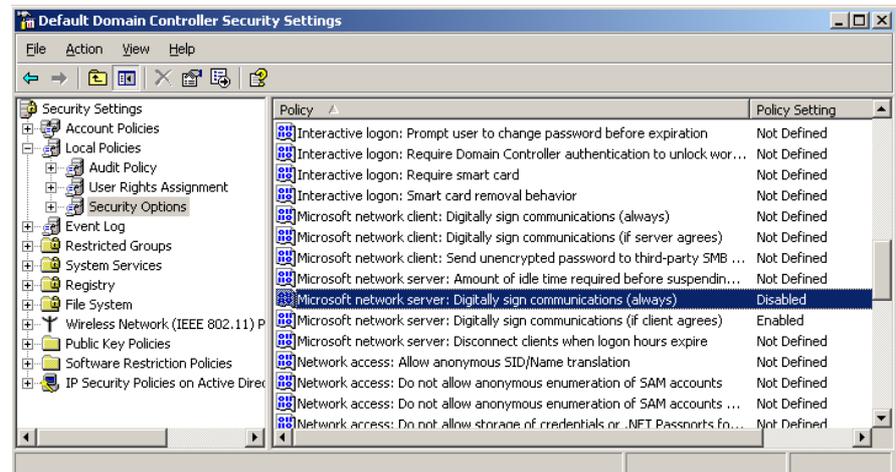
- From the edit menu, select properties, server.
- In the server properties window, modify the description field.
- Click OK or accept to regenerate the server OPT file.

The PC can not connect to a Windows 2003 Server

Modify the Default Domain Controllers Policy:

Security Settings/Local Policies/Security Options/Microsoft Network Server:
Digitally Sign Communications (always) = Disabled

And reboot the server



Can't connect to network drive

After pressing the „d“ key, the following error appears:

Error 53: The computer name specified in the network path cannot be located.

```
Error!

Can't connect to network drive
Please inform your administrator or press any key to reboot

Es konnte keine Verbindung zum Netzwerklaufwerk aufgebaut werden.
Bitte informieren Sie ihren Administrator oder drücken Sie eine
Taste um den PC erneut zu starten.

Press [d] for more informations.

Error 53: The computer name specified in the network path cannot be located.
EXPLANATION: The server that you specified does not exist or has not been
started.
ACTION: Verify that you specified the path correctly. Contact your network
administrator if you continue to have problems.
```

Possible cause:

There is a problem with name resolution.

The following steps support the name resolution.

- From the tools menu, select manage bootimage.
- In the manage bootimage window, select the master boot image and click edit.
- In the 0001 window, move to the NET directory.
- Open the LMHOSTS file with a text editor.
- Create an entry for the installation server, along with its IP address.
- Save the LMHOSTS file.
- Also in the NET directory, create a copy of the LMHOSTS file and rename it to HOSTS.
- Close the 0001 window.
- In the manage bootimage window, click generate and close the boot image management.