CA Client Automation

OS Installation Management Administration Guide

12.9

CA Technologies Product References

This documentation set references to the following CA products:

- CA Advantage® Data Transport® (CA Data Transport)
- CA Asset Intelligence
- CA Asset Portfolio Management (CA APM)
- CA Business Intelligence
- CA Common Services™
- CA Desktop Migration Manager (CA DMM)
- CA Embedded Entitlements Manager (CA EEM)
- CA Mobile Device Management (CA MDM)
- CA Network and Systems Management (CA NSM)
- CA Patch Manager
- CA Process Automation
- CA Service Desk Manager
- CA WorldView™

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 6: OSIM Images Using ImageX and Ghost32 113

Chapter 7: WinPE Boot Images Support 137

Chapter 8: How to Create and Deploy OS Images using a Linux-Based Boot Image 149

Chapter 1: Welcome to OS Installation Management

This section contains information about OS Installation Management (OSIM). It is directed to administrators who design and maintain the unattended installation of operating systems for new and existing computers in their networks.

The installation of an operating system is the first step to bring up a system. Computers must have their operating systems installed after a crash caused by hardware faults or by virus attacks. In large enterprises, network administrators want to install or repair computer software without involving a technician. OSIM facilitates these tasks.

OSIM requires computers to have a preboot execution environment (PXE), so that a basic and faultless image of an operating system can be installed.

The three main steps in the initial OS installation are:

1. Pre-OS installation (for example, hard disk partitioning, system identification)
2. Installation of the OS
3. Post-OS installation (for example, domain integration, agent installation)

Additional steps can complete the installation.

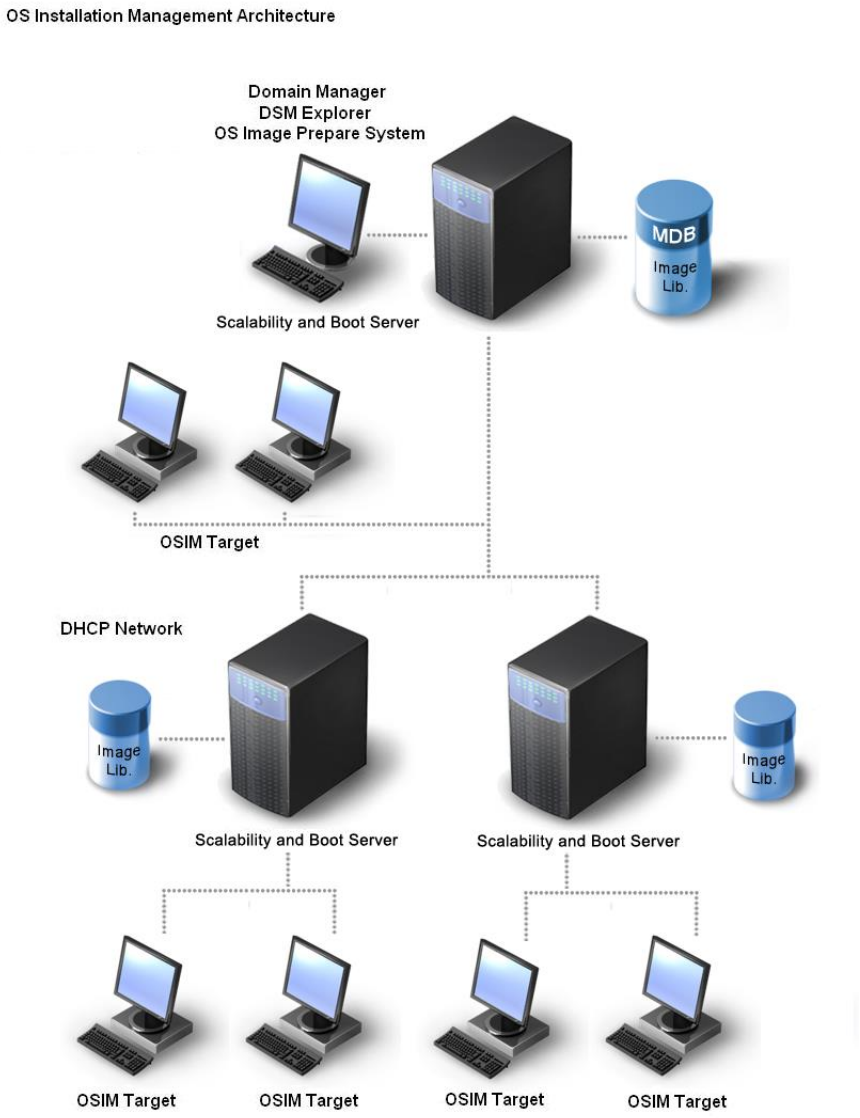
Not all systems have the same hardware, network integration, and service agents. OSIM provides boot parameters that you can use to minimize the number of procedures required to install an OS that meets the requirements of your specific environment. OSIM also provides you with a default set of parameters for all commonly used procedures and allows you to add parameters for more flexibility. In addition to this, OSIM enables you to manage and modify all boot parameters and configurations.

This section contains the following topics:

OS Installation Management Architecture

This chapter describes the architecture of OS Installation Management (OSIM).

The following picture shows the OSIM architecture.



Domain Manager

The domain manager is the central site of the OS Installation Management.
The domain manager contains all information about target computers, boot servers, boot images and OS images. It initiates and controls all the actions in the OS installation network. The information is kept in a database named MDB. The front ends in the distributed networks are the scalability servers, including the boot servers.

DSM Explorer

The DSM Explorer is the graphical user interface (GUI) for the domain manager. The DSM Explorer offers a comprehensive set of methods to drive and control the entire OS installation management network.

Image Prepare System

The Image Prepare System (IPS) is used to build the images and register them at the domain manager. The OS Installation Management differentiates between OS images and boot images. An OS image includes all OS files needed for the unattended OS installation at the target. A boot image is used to prepare a target for the OS installation and initiate the OS installation. Both types of images can be built on an Image Prepare System, and the images can be customized by adding boot parameters, default values, or procedures.

Boot Server

The OS installation works with so-called boot servers that can be set up in the network.
"Boot servers" are a functionality of scalability servers.
The boot server (BS) stages the images for all the attached PXE targets together with the boot parameters and the installation request for the different target systems. The boot server includes a PXE and a Trivial File Transfer Protocol (TFTP) service, which respond to PXE-boot requests from targets.

Note: The scalability server provides a point of buffering (staging) and resilience between end systems (agents) and a domain manager. The scalability server provides for massive scalability. Rather than having all the individually managed end systems (agents) communicating directly with a single domain manager, the load can be shared across multiple scalability servers.
For example, software packages can be staged on a scalability server before downloading to end systems, and inventory can be stored at the scalability server before being uploaded to the domain manager.

Supported Operating Systems

For the most recent list of supported versions, see the Compatibility Matrix.

Chapter 2: How to Use OSIM

This chapter describes the most common tasks required to install operating systems on target computers that have OSIM installed:

- Install a Domain Manager (Including a Boot Server) (see page 16)
- Create and Register OSIM OS and Boot Images with the OS- Image Prepare System (see page 23)

  The major step for creating boot images is described in different chapters depending on the approach you are taking:

  - Using Linux-Based Boot Image (see page 149)
  - Using Windows PE (see page 137)

  The major step for creating ImageX and Ghost-based OSIM OS images is described in different chapters:

  - Using ImageX (see page 113)
  - Using Ghost images (see page 113)
- Viewing OSIM Images in the DSM Explorer (see page 30)
- Detected and Registered OSIM-Images (see page 30)
- Make Your Target Computer PXE Enabled (see page 31)
- Make the New PXE Computer Managed (see page 32)
- Manage OS Installations (see page 36)
- Multiple Boot Servers (see page 49)
- Work with MS Shares or with TFTP (see page 54)

Note: For detailed information on installation and migration, see the *Implementation Guide*.

Install OSIM

You must use one of the following procedures to install OSIM:

- Express Manager Setup (Windows only)
- Custom Setup

To install OSIM using the Express Installation Setup on a Windows server

1. Start the Client Automation setup from the installation medium. Follow the installation wizard until you reach the installer window "Select Product Functionality".
2. Select "Software Delivery" and click "Next".

   Note: This entails the installation of the OSIM plugin at the domain manager.

   The "Select Installation Method" window opens.
3. Select "Express Installation" Setup.

   The domain manager, the Image Prepare System (see page 71), and a boot server are installed.

   Note: The Image Prepare System is running on Windows only.

To install OSIM using Custom Installation

1. Start the Client Automation setup from the installation medium. Follow the installation wizard until you reach the installer window "Select Product Functionality".
2. Select "Software Delivery" and click "Next".

   Note: This entails the installation of the OSIM plugin at the domain manager.

   The "Select Installation Method" window opens.
3. Select "Custom Installation".

   The "Select Components and Features" window opens.
4. From the components and features list, select the following options:
   a. Select the Explorer for a comfortable and GUI-based handling of images, target computers, and boot servers.
   b. Select Packaging Tools, Image Prepare System (see page 71).

      Note: Do not select any of the other packaging tools.
   c. Select "Scalability Server".

      The installation of a scalability server entails the installation of a boot server.

d. If you need share access, click the Boot Server tab of the Configure Scalability Server window to specify the boot server configuration details.

   In the boot server configuration window, click "Enable support for Windows network shares".

   The boot server will create read-only network shares using the SMB protocol.

   If not, TFTP access is used (default method).

   Note: Ghost and ImageX images need share access.

This is all you need to install the OSIM plugin at the domain manager, the Image Prepare System (see page 71), and a boot server.

Disable or Enable the Boot Server Process

When you install a Client Automation domain manager or scalability server, OSIM boot server is also installed by default on the same computer. If you do not want the computer to act as a boot server, you can disable the boot server process. You can enable the same when required.

- To deactivate and disable the boot server process, use the following commands:

```
caf stop sdmpcserver
caf disable sdmpcserver
```

- To enable and activate the boot server process, use the following commands:

```
caf enable sdmpcserver
caf start sdmpcserver
```

Note: The command must be executed at the boot server.

TFTP and Share Access

Boot servers do not provide share access to boot and OS images after a default installation. Target computers access the OS and boot images on the boot server through a special secure TFTP protocol. However, some OSIM OS types like Ghost and ImageX images need share access. Also, the installation of large OS images like Windows Vista and Windows 2008 are faster from shares.

To switch a boot server from tftp to share access, execute the following command on the boot server:

```
sdbsswitch -s
```

To switch a boot server from share to tftp access, execute the following command on the boot server:

```
sdbsswitch -t
```

Support for IPv6

OSIM relies on PXE, which in turn relies on IPv4. Therefore, using OSIM requires a network infrastructure supporting IPv4 as well. This refers to routers and network services such as DHCP, DNS, and WINS. The boot server also must support IPv4. All other systems, including the operating systems installed using OSIM, are not required to support IPv4.

Upgrade Considerations

After you upgrade the OS Image Prepare System, domain manager, and boot servers, update the OS and boot images to use the latest image templates.

Important! When you perform an upgrade or apply a patch, the impacted OS image must be updated to be compatible with the new Client Automation release. When you update an existing OS image, ensure that the value of the changed boot parameter, if any, should have the new value after the Update OS Image wizard is completed. If not, change the value manually to new one supported by OS Image.

Updating Customized Files

If you have customized the pre-installation files, post-installation files, or auto-answer files in the existing OS or boot image templates, you must update the customized files using *one* of the following methods:

- Merge the custom changes to the files in the new template and then update the existing images.
- Update the existing images with the current template and then merge the custom changes to the updated image in the image store. The custom files have "-save" appended to the original file names in the updated image. In this case, you must manually merge the custom changes to each updated image.

Important! While updating the images, you may lose the customized parameter definitions if they have been updated or deleted in the new template.

More information:

Update OS Images (see page 19)
Update Boot Images (see page 21)

Update OS Images

You must update the existing OS images with the new template files so as to be completely compatible with the new release of Client Automation. The template files control the OS installation process on the target computers. You can update the OS images either from the command line or using the wizard. You can choose to use the command line if you are updating multiple OS images using a batch script.

Note: If you have deployed the same boot or OS images on multiple boot servers, update the image and create a corresponding update SD package on the Image Prepare System first. You can then deploy the updated SD package to all the other boot servers in the domain using a software delivery job.

To update an OS image from the command line

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Execute the following command:

   ```
   CreatOSImage -i <imagename> -e
   ```

   The command updates the existing image with the new template.
3. Do the following to include additional device drivers to the OS image:
   a. Copy the new or updated device driver files to the OS image store.

      The OS image is updated with the device driver files that needs to be deployed when installing the OS.

   b. Edit the <image store>\<image name>\osinfo.ini file of the image and specify the list of files you want to add or delete in the [update] (see page 87) section.

      This step helps ensure that the additional files are deployed on the target systems.

4. Execute the following command to update the OS image on the domain manager:

   ```
   registerOSImage –i <image name> -s <manager> -e
   ```

   The command updates the image on the domain manager specified with the -s parameter.

5. Execute the following command if you need an SD package for updating the OS image on a remote boot server:

   ```
   registerOSImage –i <image name> -s <manager> -e –l
   ```

   The command creates an SD package with the updated OS image. The updated package has "-update" appended to the original package name. The updated package can be used to update the OS image on remote boot servers using an SD job with the "upgrade" procedure.

6. Execute the following command to update the parameter definitions of the image:

   Important! You may lose the customized parameter definitions if they have been updated or deleted in the new template.

   ```
   registerOSImage –i <image name> -s <manager> -e –b
   ```

   The command updates the parameter definitions of the OS image in the MDB. The updates include changes to the default values, new, updated, or deleted parameter definitions.

   The OS image is now updated in the domain manager and in the boot servers.

To update an OS image using the wizard

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Navigate to Software, Boot and OS Image Library, Image Prepare System, Update OS Image in the DSM Explorer.

   The Update OS Image wizard appears.

3. Follow the instructions in the wizard to update the OS image.

   The OS image is updated and registered with the domain manager based on the options you have selected. You must now deploy the updated SD package on the other boot servers that stage the OS image that you just updated.

More information:

Make Boot and OS Images Available on Boot Servers (see page 52)
Updating Customized Files (see page 18)

Update Boot Images

You must update the existing boot images with the new template files and any content updates published by Client Automation, so as to be completely compatible with the new release of Client Automation. The template files initialize the OS installation process on target computers. You can update the boot images either from the command line or using the wizard. You can choose to use the command line if you are updating multiple boot images using a batch script.

Note: If you have deployed the same boot or OS images on multiple boot servers, update the image and create a corresponding update SD package on the Image Prepare System first. You can then deploy the updated SD package to all the other boot servers in the domain using a software delivery job.

Important! Linux-Based Boot Image has been patched up to mitigate Shellshock vulnerability (Bash Code Injection Vulnerability). Updated Linux-Based Boot Image (Customized CentOS 6.3 x64 boot files) available on a DVD separate from the product DVD.

To update a boot image from the command line

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Execute one of the following commands depending on the boot OS type:

   For WinPE20, WinPE30, WinPE20x64, or WinPE30x64

   ```
   CreateBTImages –i <image name> -e
   ```

   For WinPE and WinPEx64

   ```
   CreateBTImages –i <image name> -c <winpepath> [-w <ospath>] -e
   ```

   The command updates the existing image with the new template.

3. Execute the following command to register the boot image:

   ```
   RegisterBTImages –i <image name> -s <manager> -e -b
   ```

   The command updates boot image data in the MDB.

4. Execute the following command to update the parameter definitions of the image:

   ```
   RegisterBTImages –i <image name> -s <manager> [–v <new version no>] –e -l
   ```

   The command creates an SD package for the updated boot image and updates the boot image on the boot server. The updated package has "-update" appended to the original package name. You can stage the updated package on remote boot servers using an OS installation job.

   The boot image is now updated and is registered with the domain manager along with the updated SD package.

To update a boot image using the wizard

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Navigate to Software, Boot and OS Image Library, Image Prepare System, Update Boot Image in the DSM Explorer.

   The Update Boot Image wizard appears.

3. Follow the instructions in the wizard to update the boot image.

   The boot image is updated and registered with the domain manager along with the updated SD package.

Using the DSM Explorer for OSIM Tasks

The Image Prepare System (IPS) GUI substantially facilitates completing the OSIM tasks with wizards, guiding you through the following processes:

- Creating boot images
- Creating OS images
- Importing boot images to Client Automation
- Importing OS images to Client Automation
- Creating and managing OS installations

The IPS GUI is part of the DSM Explorer. In the DSM Explorer you manage the Boot and OS Image Library, both on the local system and on the boot servers.

Boot and OS Image Library

Boot Images

Boot images registered in the domain

OS Images

OS images registered in the domain

Image Prepare System

Create, customize, and register boot and OS images

For images which are available as Client Automation software package, browse the Software Package Library of the DSM Explorer.

The following sub-sections are available in section Image Prepare System:

Local Boot Images

All boot images created on the local Image Prepare System

Local OS Images

All OS images created on the local Image Prepare System

Wizards

Wizards to create and register boot an OS images

Boot and OS Image Staging Library

The list of all boot images on boot servers is shown in the Scalability Section. For each scalability server which also acts as a boot server there is a folder Boot and OS Image Staging Library with subsections Boot Images and OS Images.

OS Installations

Shows a list of all the OS installations of the PXE computers connected to a boot server.

New PXE enabled computers introduced to the boot server's network are automatically added to the list.

Create OS and Boot Images Using the Image Prepare System

You need OSIM boot and OS images to install an operating system on new computers or computers that have stopped functioning due to critical and irreversible failure.

To create the boot image, use the Create Boot Images wizard.

Before you can use the boot image with the domain manager, you have to register it with the Register Boot Images wizard.

With the Create OS Image wizard, you build an OSIM OS image that contains helper programs, configurations scripts, and setup files for a specific operating system.

The Create OS Image wizard displays the supported platforms and image types for OSIM. Furthermore, they are listed in this guide in the Image Prepare System (see page 71) section.

Within the DSM Explorer, add or modify boot parameters used by OSIM OS images for installation on a target computer or to have greater control over the delivered setup files and configuration scripts.

Before you can use an OSIM OS image with the domain manager, it must be registered with Register OS Image wizard.

Note: Instead of creating and registering images with wizard assistance in the DSM Explorer, you can also use the underlying commands directly. They are documented in detail in the Image Prepare System (see page 71) section.

Create OSIM Boot Images

The boot image is essential to the OSIM process. The boot image drives the unattended operating system installation on the target computer.

The default boot images created with the Image Prepare System normally meet the requirements of all supported OS images and target computers. For licensing reasons, these default images cannot be delivered by CA, but must be built by the user.

For the list of boot images used for each operating system, see the Compatibility Matrix.

The necessary prerequisites for boot image creation are described in the following sections:

Windows PE-based boot images

Using Windows PE (see page 137)

Linux-based boot images

Using Linux-Based Boot Image (see page 149)

How to Introduce OSIM Boot Images to the Domain Manager

Boot images must be introduced to the domain manager before OSIM can use them.

To introduce OSIM Boot Images to the Domain Manager, the images must be registered either using the Register Boot Images wizard of the DSM Explorer or by using the registerBTImages command directly.

- Execute the command:

  ```
  registerBTImages –i linuxpex64.2 -s myDomainManager
  ```

  The command registers the boot images at the OS Installation Manager, myDomainManager, and creates a software package named BootImages with the version number 11.1.0/00 at the associated domain manager.

- If the name and version of images must be modified, execute the command:

  ```
  registerBTImages –i linuxpex64.2 -s myDomainManager -n mybootimagename -v 1.1/11
  ```

  The boot images are now ready for deployment through Client Automation and for application in OS Installation Manager.

- If Unified Logon is not used, remote Linux and Windows domain managers may need a user name, domain, and password to allow registration of boot images. To enable automatic logon and authentication, use:

  ```
  registerBTImages –i linuxpex64.2 -s myDomainManager -u root -p password
   -d unixl://myDomainManager.xx.com
  ```

Create OS Images

To create OS images, use the CreateOSImage command.

Use CreateOSImage -x option for a list of already created OS images that are available in the Image Prepare System.

The path to the OS images in the image store is listed also. This parameter is useful when you want to make manual adjustments to an OS image. For example, the following command creates a Windows XP image from the Windows XP CD:

```
createOSImage –I myxpsp2 –o WXPP –k <product key>
```

The following command creates a Windows XP image from a Windows XP directory structure:

```
createOSImage –i myxpsp2 –o WXPP –k <product key> -s d:\i386
```

For a complete description of the CreateOSImage command, see CreateOSImage.

Add OSIM Parameters to Created Images

OSIM OS images have various parameters to control the installation process. Default OSIM OS Images have all parameters for a successful installation.

To have extended control over the installation procedure, you can add more parameters to images that have already been created. OS images must be modified to add parameters before being registered with the Domain Manager. If the OS image has already been registered, remove the image from the manager, modify and re-register the image.

When an OS image is created using the wizard or CreateOSimage command, it is, by default, stored in a directory called Image Store.

Note: You cannot remove an image from the Domain Manager as long as the image is assigned to target computers.

When adding parameters, keep in mind the following:

- CreateOSImage and RegisterOSImage look for parameters only in files listed in the [parameterfile] section in osinfo.ini.
- CreateOSImage -x shows all created images and their location.
- CreateOSImage -p <imagename> shows the properties of one image including the files with parameter in it.
- If you insert a parameter $parametername$ in one of the files without defining the parameter in default.ini, RegisterOSImage will take over the new parameter with default properties (type=string, max length= 255, default value = “”, comment = file where it was found).
- You can define parameter type, default values, comments in the default.ini of each OS image. The locations of the default.ini files are under the paths listed with CreateOSImage -x.
- You can check your parameter changes with RegisterOSImage -i <name> -t. This command will also extend default.ini with default definitions of new parameters not yet defined.

For more details, see default.ini (see page 91).

Example: Add OSIM OS Image Parameters

Important! Only administrators with a good understanding of .inf and default files should change parameters or add new ones.

For example, you create an OS image named myImage and you want to add parameters setting the Language variable of the regional settings in the auto answer file of the OSIM OS image, myImage.inf.

The parameter that should drive this variable is named "localeID" and its default value should be 1033.

To add a parameter to the image

1. Load …\ManagedPC\CAMENU\myImage.inf into an UTF-8 enabled editor, for example, Notepad. The CAMENU directory is in parallel to the IMAGES directory.

   Note: See createosimage -x for the exact path to the myImage directory.

2. Go to the [RegionalSettings] section in myImage.inf.
3. If there is already a value for Language, change it to the following:

   ```
   Language=$localeID$
   ```

   Otherwise, add this line to the section.

4. Save the file and close the editor.
5. Load …\ManagedPC\IMAGES\myImage\default.ini into a UTF-8 enabled editor, for example, Notepad.

   See createosimage -x for the exact path to myImage.

6. Go to the [Default] section, and insert the following line in this section:

   ```
   localeID=1033
   ```

7. Add the following new section at the end of the file:

   ```
   [localeID]
    Type=MapListExt
    MaxLength=128
    Comment=Language/locale to be installed
    Trans=yes
    item=5124 Chinese_Macau
    item=1030 Danish
    item=1033 English US
    item=2057 English UK
    item=1036 French Standard
   ```

   For details about the parameters, types, and attributes, see Default.ini Parameters.

8. Save the file and close the editor.

Extend the Image with Special Drivers

Older operating systems do not include device drivers for new computer hardware. Network access especially will not work when using unsuitable drivers for the network adapter. In this case, the agent cannot connect to the manager and the installation seems to hang.

If the OS does not have sufficient drivers, ask your hardware vendor for the correct device driver that is suitable for your OS image. Introduce the driver into the OS image on the Image Prepare System (see page 71) before you register and distribute the image.

For the most recent list of supported versions, see the Compatibility Matrix.

Extend createOSImage with New OS Image Types

The IPS has a set of OS image templates. These templates can be viewed using the CreateOSImage command. All available images are defined in the Template.ini, which controls the CreateOSImage command.

You should be familiar with the process of creating new image types and installation templates before you attempt to create them.

You can add a new image type by adding a [ostype] section including the keys for specification. See the key descriptions following.

Template files in the os-template\camenu\<ostype>.xx and in the os-template\images\<ostype> also belong to an OS type. The assignment is defined in the template.ini file.

For more information, see Template.ini (see page 97).

Introduce OSIM OS Images to the Domain Manager

To introduce OSIM OS Images to the Domain Manager, the images must be registered by using the registerOSImages command. For example:

```
registerOSImage –i myOSimageName -s myDomainManager
```

For complete information about the registerOSImage command, see RegisterOSImage (see page 80).

How to Register OSIM Images to Another Domain Manager

To register OSIM images to other domain managers, do *one* of the following:

- Use remote registration to a remote domain manager (see page 29)
- Export an SD package from the domain manager and import it into remote domain manager (see page 29)

Remote Registration from Image Prepare System to Remote Domain Manager

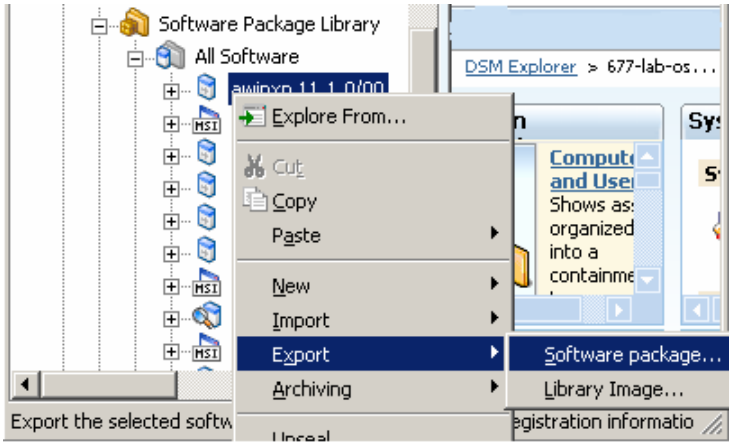
To register an OS image to another domain manager, use remote registration from Image Prepare System (see page 71) to the remote domain manager:

```
registerOSImage
        -i image
        -s "remote domain manager"
        -u user
        -p -password
        -d security_provider://"remote domain manager"
```

Export / Import Images

To register an OSIM image (boot image or OS image) with the help of exporting / importing SD packages:

1. Export the OS SD package to a directory as shown in the following screenshot:



2. Transfer the directory to the remote site.
3. Import the OS SD package or Boot Image SDpackage at the remote site using the wizard or one of the following commands:

   ```
   registerOSImage -w directory -s domain_manager
   ```

   ```
   registerBTImages -w directory -s domain_manager
   ```

Viewing OSIM Images in the DSM Explorer

You can see the created and registered images in the All Software folder in the Software Package Library.

All images created and registered at the Image Prepare System (see page 71) are displayed in the All Boot and OS Images sub-folder of the Software folder.

When you select an image, all of its parameters are displayed. If applicable, you can edit the default values or reset them to their default settings.

You can change or reset the default values with the DSM Explorer as long as the image is not assigned to a target computer.

Detected and Registered Images

The boot server looks for OS and boot images stored in their image library and reports the images to the OSIM domain manager. If the boot server and Image Prepare System (see page 71) reside on the same system, the images that have been created but have not been registered are displayed as "detected" images in the All Boot and OS Images folder in DSM Explorer.

After registration with RegisterOSImage, the detected image can be used for OS installation orders.

PXE-enabled Target Computers

Enable Network Startup in BIOS

Network Boot is done through the PXE protocol. All computers will provide BIOS settings for the boot of the computer. Typically, the network adapter on the motherboard and the BIOS also provide the Network Boot option. To use the computer with OSIM, Network Boot should be the first boot device. The following screenshot shows Network Boot selected as the first boot device in the BIOS Setup Utility:

Old computers without an onboard network adapter need a network card with support for Network Boot. These cards typically have separate configuration tools.

Reboot the Computer to Broadcast a PXE Request

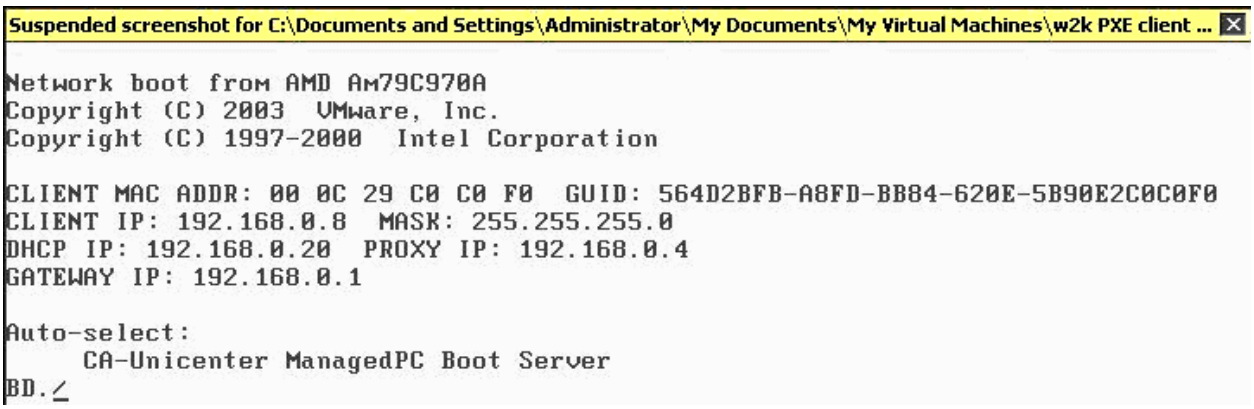
When a computer starts up, it broadcasts a PXE request in the subnetwork.

Boot servers watch for PXE requests from target computers. If a boot server detects a pre-configured number of retries from a target, it will answer the target request. This applies to all boot servers in the subnetwork. The target will use the first answer received and connects to the boot server that sent it.

The following information will be displayed on the target:

- The MAC address of the target (Client MAC ADDR)
- The IP address and network mask the target sent from the DHCP server
- The IP address of the DHCP server
- The PROXY IP which is the IP address of the selected OSIM boot server
- The IP address of the default gateway sent from the DHCP server
- The message that an OSIM boot server was selected (CA-Unicenter ManagedPC Boot Server)

The following screenshot shows the information that will be displayed on the target:



How to Manage a New PXE Computer

You can add OSIM OS installation support to a DSM managed computer either manually or automatically.

Manually: Add OSIM Support Before the Target Is Detected by a Boot Server

The OS Installation method for a target computer starts an OSIM wizard that adds an OS installation order (Planned or Activated) to the target.

You have to choose a boot server in one of the dialogs.

If you do not know to which subnetwork and boot server the target will be reported, select any of the boot servers.

The target will be moved automatically after another boot server reports the target for the first time.

Note: In this case, it needs one target boot to catch up the target and another boot to start the OS installation. If the other boot server does not have the required OS image in its image store, the OS installation order will fail.

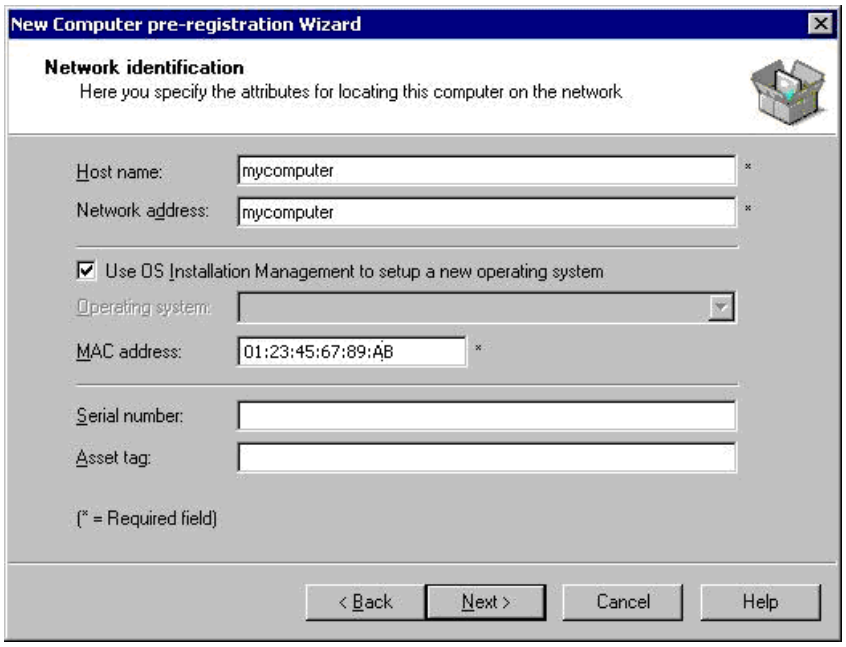
To start the OSIM wizard

1. Navigate to Computers and Users, All Computers in DSM Explorer.
2. Right-click the target computer that you want to manage, and click OS Installation.

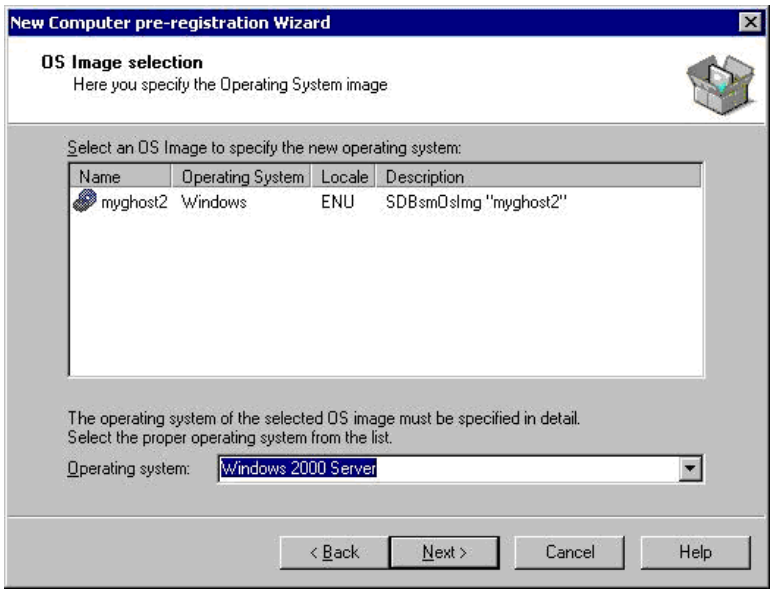
Predefined PXE Computer Plugged In Later

Right-click All Computers and in the context menu that appears, select New Computer. A wizard is displayed; follow the instructions in the wizard.

You must know the MAC address of the computer. OSIM can identify the computer by the MAC address, when it starts and connects to the network. The following screenshot shows an example MAC address specified for a new computer in the New Computer pre-registration Wizard:



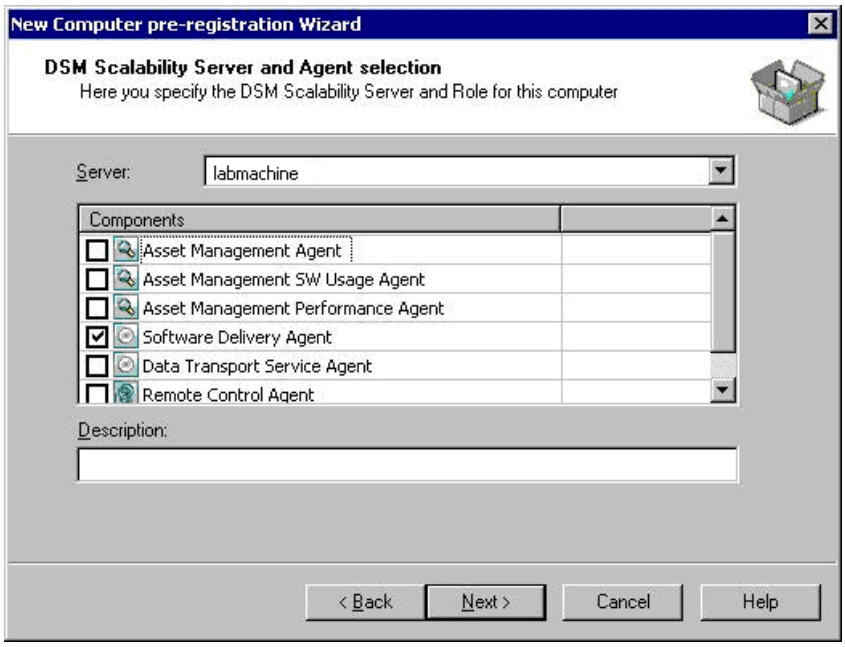
Select an OSIM OS image from the list. If the image covers more than one software delivery type, select the appropriate delivery type from the drop-down list as shown in the following screenshot:



Select a boot server as shown in the following screenshot:

Note: If the target is picked up by another boot server, the target will be automatically moved to that other boot server.

Note: Activated OS installations can fail if the OS image is not available in the image store of the responsible boot server.



Note: If you do not select setup OS installation later (edit Boot Parameter first) in the following dialogs, the OS installation order becomes active immediately.

Automatically: OSIM Support After Target Is Detected by a Boot Server

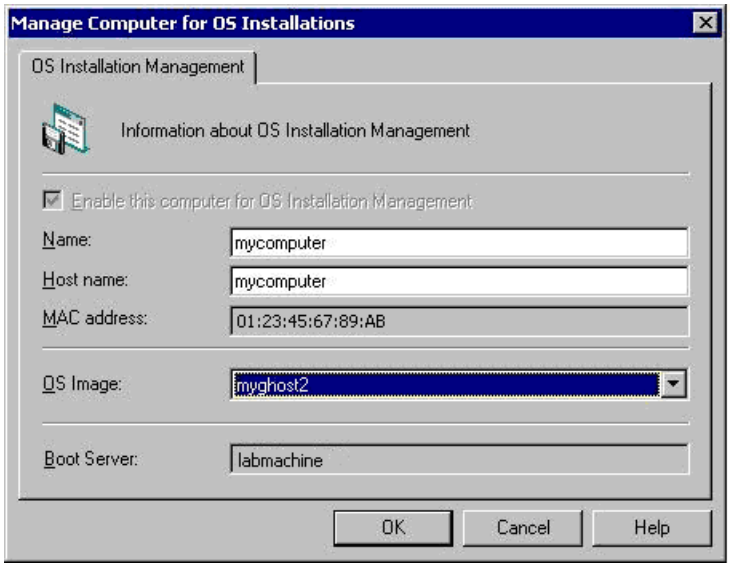
Make a computer PXE-enabled (Network Boot) and restart the computer.

After that, the computer is picked up by a boot server and reported to the domain manager.

In the DSM Explorer, the computer will be displayed with its MAC address in All Computers, Group Details, OS Installations. The state of that computer is "unmanaged".

To make this computer OSIM-managed, right-click the MAC address and select "Manage (unmanaged)".

If you choose an OSIM OS image from the drop-down list, the following dialog will assist you in creating a planned OS installation.



The OS installation can be activated, and when the target is restarted, the installation will start.

Manage OS Installations

Use *one* of the following methods to assign an operating system image to a named PXE computer:

- Copy/Paste
- Drag/Drop
- Manage

For example, copy the OS image myghost2 and paste it on the computer abcde.

1. Select an image from Software, All Boot and OS Images, for example, myghost2.
2. Right-click and select Copy.
3. Select OS Installation Parameters for the target computer in Computers and Users, All Computers, for example, abcde, right-click and select Paste, OS Image to Install.

A new planned OS installation will be created on the target computer. You can view the installation parameters in the OS installation Parameters node under the target computer in DSM Explorer. You can change parameters before activating the OS installation.

Boot Parameters

You can add or modify boot parameters of OSIM OS images to have greater control over the delivered setup files and configuration scripts. You can modify the boot parameters at OS Image or OS Installation level. Modifying the boot parameters at the OS Image level, sets the modified value as the default value for all deployments based on the OS image. If you modify the boot parameters at the OS Installation level, the modified values are only set for the target machine.

Boot Parameter Settings

The following section details the boot parameters for OS images:

OSUser

Specifies the registered owner of the Windows computer that is created during installation.

Default: OSUser

OSPasswd

Specifies the password that is associated with the OSUser.

Default: *default*

Note: For VMware ESXi, RH5XPE, and RHEL6 the default password is *123Default-*

HostName

Specifies the host name for the installed system.

The HostName is set with the same name as the Host Name entered in the New Computer Pre-registration wizard.

To enable OSIM on a computer where the given HostName is not OSIM-compliant, you have to change the HostName by selecting the computer in the DSM Explorer, right-clicking Properties, and selecting the OSIM tab.

Enable the computer for OSIM and type in a correct, OSIM-compliant host name.

For example, change "Mein schöner + großer Computer" to "MeinSchoenerGrosserComputer".

The Host Name can be changed later in the OS Installation parameters, but only as long as no DSM agent has reported the computer to the DSM manager.

The installation of a VMware ESXi Server does not register the host name in DNS. The host name is only defined locally on the ESX Server. For remote access, such as by the VMware vSphere-client, you have to use the assigned IP address instead. For VMware ESXi, it works only when the value of bootproto is "static". If you need a custom host name:

- Specify the value for NetProtocol as Static
- Specify appropriate values for the following parameters:
  - NetDefGateway
  - NetIPAddress
  - NetSubNetMask

OSImageNFS

The parameter presetting is the name of the NFS share as specified during the creation of the OS image. You can change the value before initiating the OS installation.

BootServerNFS

If the OS Image uses a central NFS share, this parameter defines the name or IP address of the server providing the NFS share. If the value is empty, the assigned boot server is assumed to be the NFS server.

Note: If you specify a name for a VMware ESXi installation, the name must be a full qualified name (FQN).

Interface

Specifies the name of the network device and defaults to vmnic0. For VMware ESXi this parameter (vmnic0, vmnic1 …) refers to the uplink device for the virtual switch that is created for the service console. This parameter (eth0, eth1 …) specifies an Ethernet device for the installation.

NetProtocol

Defines whether to use static or DHCP enabled IP address. When you want to use static IP address, change the default setting from DHCP to static.

The following boot parameters need special consideration when static IP addresses are used.

NetIPAddress

Defines the IP address of the machine to be installed.

NetDefGateway

Designates the default gateway as an IP address.

NetSubMask

Specifies the subnet mask for the installed system. The default value is *255.255.255.0*

DNSServer

Designates the primary name server as an IP address. The IP address of a secondary name server can be added with a preceding comma.

DNSSuffix

Defines a search list for host name lookup. Use spaces for separating multiple suffixes.

BootSpecial

Defines the name of the boot image.

For Windows installation, the value of this parameter can be a 32 or 64-bit WinPE image. The default value is WinPE 3.0 (AIK 2.0) 64-bit.

OS installations like VMware ESXi, RHEL, Citrix XenServer, and Kubuntu require Linux-based boot image.

Note: This parameter must be set if the job inherited the name of a WinPE boot image from previous Windows installations.

InstallDrive

(For Linux-based boot image only) Specifies the install device name. The user specifies this boot parameter to determine the local disk for OS installation. The default value is empty and the first available local disk is used for OS Installation.

InstallAgent (yes|no)

This boot parameter is used by the OS image and the OSIM manager to determine whether to install a DSM agent and to use the agent registration to signal completion of the OS installation. It defaults to Yes.

For Hypervisors (VMware ESXi, Citrix Xenserver), set this parameter to No. The installation status does not change to current (completed) as there is no native DSM agent available.

- If the value is yes, OSIM installs a DSM agent on the target. The state of the OS installation becomes Current after the DSM agent is installed successfully.
- If the value is no, OSIM will not install the DSM agent and the state of the OS installation becomes Current after the first start from a hard disk after a sequence of Network Boots.

SignalCompletion

SignalCompletion signals the completion of the OS installation in addition to the InstallAgent parameter.

When the SignalCompletion parameter is set to Yes, the completion of OS installation sends a TFTP-based signal to the host boot server. The host boot server signals OS installation completion and changes configuration state to "current".

CA ITCM agent installation is not supported on VMware ESXi, VMware ESXi 5.1, and Citrix XenServer hypervisors. Hence, InstallAgent parameter is set to No by default for these operating systems.

The following table describes the SignalCompletion and InstallAgent settings for the four use cases:

InstallAgent Value	SignalCompletion Value	Status of the OS Installation Job
Yes	Yes	Job is marked as current after successful OS installation.
Yes	No	Job is marked as current on the OS that have agent support in CA ITCM. For other OS like hypervisors, the job remains in installing state.
No	Yes	Job is marked as current after the successful OS installation.
No	No	Job is marked as current after the Boot Image downloaded.

Domain, DomainPasswd, DomainUser

If the parameters are set with values for a domain, the OS installation will add the target to the given domain. OSIM will use the DomainUser and the DomainPasswds parameters to get access to the domain controller.

VLANID

An optional VLAN ID can be passed though to configuration. This is an integer value between 0 and 4095. This is applicable for ESXi.

ProductId

This boot parameter specifies the ESXi license key as used during installation. If not set (or set to 000-000000), then ESXi will be installed in evaluation mode.

Kubuntu OS image

The following boot parameters require special consideration for Kubuntu OS image:

OSUser

Specifies the user name that is created during installation. OSIM uses a non-privileged user name but not a root account. Verify that you do not use reserved user names, such as root and wheel.

Default: osimuser

OSImageShare

Specifies the exported shared directory name from where the OS files are accessed for the OS installation.

Country and Language

Specifies the country and language for OS installation. The installer uses these two parameters to determine the language of the OS. If the combination does not form a valid locale, the installer automatically selects a locale that is valid for the selected language. For example, Country=Germany and Language=German causes the installer to install the OS in German. For a list of supported locale, see the Ubuntu documentation.

Keyboard

Specifies the keyboard layout, which is independent of the country and language setting.

ServerHostProtocol

Specifies the protocol that the installer uses to access the boot server and Ubuntu repository for configurations and packages. Use either FTP or HTTP.

Note: Verify that you use the same method for OS deployment as the server you have configured on the boot server. For example, if you have configured an FTP server on the boot server, during the deployment of OS to the boot server, the SD agent creates an FTP share. If you then create an OSIM job using the HTTP method, deployment fails with the error "Failed to retrieve preconfiguration file".

ServerHostXTP

Specifies the name or IP address of the server providing the Ubuntu FTP or HTTP accessible repository.

Default: IP address of the boot server

Note: Verify that you have exposed the mirror repository share with the protocol you have selected. For example, if you select FTP here but you have only configured HTTP access to the share, the installation displays a warning message: “Bad archive mirror.”

ServerHostRoot

Specifies name of the root entry of the repository for Kubuntu images

Example: /ubuntu

Citrix XenServer Boot Parameters

The following boot parameters require special consideration for Citrix Xenserver:

GuestDisk

(Available for XenServer Deployment) The name of storage device to store the guests. This parameter supports multiple formats like logical name sda, sdb or SANID or WWID or disk/by-id/scsi-SANID or disk/by-id/scsi-WWID.

Warning! in case of target with both local and remote (SAN) disks, supply of the logical names like sda, sdb etc. may not be persistent for OS installation.

InstallBase

(Available for XenServer Deployment) The name of the folder where the installer must look for the packages. Use 'packages.main' for default install. If Linux support is required, ensure packages.linux is copied to the parent folder and choose the empty folder name.

InstallGuestOnPrimary

(Available for XenServer Deployment) This parameter specifies whetheher a guest storage repository is created on the disk where the control domain is specified or not.

SAN ID Parameter

The SANID boot parameter is added to the OS images to identify a specific target related to a particular SAN logical unit number (LUN) or Disk. This provision ensures that the specified LUN or Disk is explicitly selected. SANID boot parameter holds the identifier value that is associated with the specified Disk or LUN like World Wide Name (WWN) or World Wide Identifier (WWID) based on Network Address Authority (NAA) format.

The supported formats (with example values) for SANID are:

```
value
    naa.60a9800064762f34694a6e3651697759
    60a9800064762f34694a6e3651697759
```

For backward compatibility, support for the SANID value in the *attribute=value* format is available as shown below:

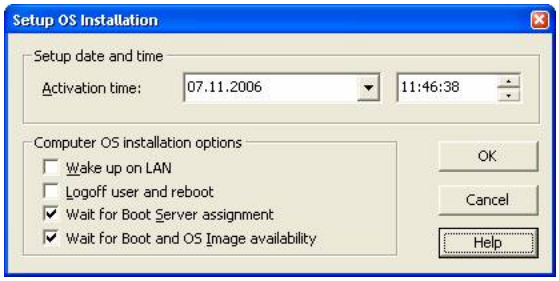
```
    TargetLUN=naa.60a9800064762f34694a6e3651697759
```

The support for providing the SANID parameter using EUI format is dropped from this release.

When the SANID value is empty, the installation scripts use the first available local disk for the installation per the existing behavior. Also, this disk must be the first disk in the BIOS boot sequence.

Activate the OS Installation

Activate the OS installation by right-clicking the system entry and selecting Activate Installation (planned) from the context menu. The Setup OS Installation dialog appears, providing date, time, and installation options. The following screenshot shows the Setup OS Installation dialog with the activation time and the installation options:



Wake-On-LAN in Boot Server Network Segments

Waking up a target computer can be selected for each OS installation request.

If Wake-On-LAN is selected, the responsible boot server will wake up the target computer when the activation time is reached and after the OS installation order is scheduled on the boot server.

To wake up a target the boot server needs information about the target machine's IP subnet. The boot server collects subnet information from the reports the Agents send to the Scalability Server.

If a target is in an IP subnet remote to the boot server please make sure at least one Agent connected to that subnet already has registered to the Scalability Server the boot server is running on.

Note: A target computer has to be enabled for Wake-On-LAN. This setting can be enabled in the BIOS power management settings. For more information on enabling Wake-On-LAN, see the documentation for your computer.

Logoff User and Restart

To start an installation, you must restart the target computer. If a computer cannot be restarted, you can force a restart with the OS Installation option.

If Logoff User and Reboot is selected, the target computer starts a logoff and restart procedure at the given activation time. It is the installed agent which starts the target computer.

If a user is logged on to the target computer, the agent will raise a message and a five-minute countdown starts.

Each time the user clicks the Postpone button, the countdown restarts with a duration of 30 minutes.

Wait for Boot Server Assignment

If the target is not assigned to a boot server, the activation of the OS installation waits until the target is picked up by a boot server and reported to the domain manager.

Wait for Boot and OS Image Availability

If the target is assigned to a boot server, the activation of the OS installation waits until the required OS and Boot image is available on the boot server.

States of the OS Installation Job

- After activation the OS installation job will change its state from planned to activated.
- At the activation time the job will be sent to the Boot server and the state becomes pending.
- When the target started the installation (it loads the first boot image) the job state becomes installing.
- Depending on the $InstallAgent$ job parameter the OS install job becomes the state current after the DSM agent registered successful to the manager.

Jobs can be stopped or aborted depending on the state pending or installing.

Delete an OSIM-managed Computer from OS Installation Management

OS Installation Management now presents a method to completely remove a computer from the OSIM database. This method also replaces the former method to delete an unmanaged or ADS managed computer: Delete (unnamed / ADS managed) (see page 45).

To delete an OSIM-managed computer from OS installation management

1. Navigate to Computers and Users, All Computers, {Group Details}, OS Installations.

   The right pane shows all the managed and unmanaged computers in the All Computers group.

2. Right-click the computer you want to remove and click Delete from OS Installation Management.

   When the operation completes successfully, the selected computer will be removed from the list of OS installations.

Re-install the Current OS

Depending on the planning status of the affected system, you can reinstall the system in different ways.

To check the situation, open All Computers | <computername> | OS Installation Parameters. The assigned configurations and their states appear.

As you review the status of the affected system, determine which of the following scenarios best fits your situation and follow the steps provided:

- There is a current configuration assigned to the affected computer but no scheduled or planned configuration

  To reinstall the current configuration, follow these steps:

  1. Right-click the system entry and select Reinstall (Current) from the context menu that appears.
  2. Click OK on the Setup and Activation Time window to start the reinstallation immediately.

- There is a scheduled configuration assigned to the computer

  You have the choice either to proceed with the scheduled configuration, which you can accelerate by switching the activation time to Immediately, or to reinstall the current configuration.

The steps for each option follow.

To accelerate the scheduled configuration in Activated status where an activation time is set too far in the future

1. Right-click the configuration entry and select Stop Installation (Activated/Pending) from the context menu.
2. Wait until the scheduled configuration reports a Stopped status (shows the (Stopped) icon).
3. Right-click the configuration entry and select Renew Installation (Failed/Stopped) from the context menu.
4. Click OK in the Setup OS Installation window to schedule the installation request for immediate execution.

Use Groups to Make Handling Easier

The OS Installation can be part of a DSM group:

1. Create a DSM group and copy DSM computers into the group.

   You can use this group to change OS installations. You can also create a dynamic group by associating a query to the group. For example, if you want to use a group of computers where the OS image winxp1 is installed, define a query and name it. The query is then assigned to the group. To create a dynamic group, follow these steps:

   a. Click the Query tab on the New Group dialog.

   b. Select the query you want to associate with the group and click OK.

      The new group is added under the Computers and Users node. The new group does not have any computer associated with it.

   c. Right-click the new group and click Evaluate Now.

      The new group is associated with the computers resulting from the selected query.

2. Select the OS Installations folder under the new group.

3. On the right pane, select all the computers, right-click, and select Edit to change the selected parameter.

   The Edit OS Installation Parameter dialog appears, which lets you edit the selected parameters.

Chapter 3: OSIM Boot Server

This section contains the following topics:

Multiple Boot Servers

This section describes having more than one boot server in a domain and how to use this feature.

This section contains the following topics:

Manage Additional Boot Servers

You can now install additional boot servers in a domain using the following alternatives:

- Install from installation media
- Install with Software Delivery

To install an additional boot server from the installation media

1. As the boot server is part of a scalability server, install a scalability server.
2. Choose between access with or without shares.

   For more information, see Install OSIM (see page 16)

To install an additional boot server with Software Delivery

1. If the boot server system has a DSM agent already installed, use the DSM Explorer and install the Scalability Server package.

   Select Scalability Server from the Software Package Library and drag it on the new boot server system in the "All Computers" folder.

2. When the boot server has been installed successfully, it will report itself to the OSIM plugin of the manager.

   Note: The default installation prepares the boot server for TFTP access. The boot server is enabled and active after the installation.

Multiple Boot Servers in an IP Subnetwork

With a new scalability server, a boot server is also installed. The new boot server does not take responsibility for target computers already assigned to another boot server in the subnetwork. Each boot server is responsible for a specified set of target computers only. The DSM Manager determines the set of target computers.

All boot servers observe PXE requests in their subnetwork, but they immediately reply only to requests of machines belonging to their own set of target computers. Requests from other machines are answered only after a certain number of retries and after a certain amount of time. The number of retries and the time to wait can be configured. More than one boot server may send such a default response.

A target machine receiving one or more default responses will choose one of the replying boot servers and establish a connection. This usually is the boot server that replied first.

A new target (not yet assigned to a boot server) will be fetched by that boot server.

A target already assigned to a different boot server will be reassigned to the new boot server. If there is an activated or pending installation order for the target and the required OS image is not available at the new boot server, the installation order is canceled.

The following configuration policy parameters under the DSM, ScalabilityServer, OSIM, ManagedPC, Server node change the behavior of a boot server with respect to other boot servers in the IP subnetwork:

Use ACL

Indicates whether the boot server uses an Answer Control List (ACL). Valid values are:

0

If Use ACL is set to “0”, the boot server must be the only boot server in its IP subnetwork because it answers all PXE requests immediately.

1

If Use ACL is set to “1”, the boot server immediately answers PXE requests of assigned targets only, that is, the boot server uses an ACL.

PXE requests from other target machines will be answered only after a certain number of retries has been seen and after a certain amount of time has elapsed since the first request.

2

If Use ACL is set to “2”, the boot server immediately answers PXE requests of assigned targets only, that is, the boot server uses an ACL, but it does not answer to unknown targets.

Default: “1”

Discovery retries before answer

Number of retries before a boot server sends a default reply to the PXE request of a target not assigned to it.

Limits: “1” to “4”

Default: “3”

Note: Only evaluated if "Use ACL" is “1”.

Discovery timeout before answer

Number of seconds to wait before a boot server sends a default reply to the PXE request of a target not assigned to it.

Limits: “3” to “56”

Default: “10”

Note: Only evaluated if "Use ACL" is “1”.

The following table shows corresponding DiscoveryRetriesBeforeAnswer and DiscoveryTimeoutBeforeAnswer values:

DiscoveryRetriesBeforeAnswer	DiscoveryTimeoutBeforeAnswer
1	2
2	8
3	20
4	40

Any of the boot servers in an IP subnetwork can take responsibility for a PXE target. Therefore, it is recommended you store the needed OS and boot images on any of the boot servers in the subnetwork.

Delay PXE Response for Known Target:

The number in seconds for delaying the PXE Response from Boot Server to known (managed) PXE target.

```
Only if UseACL = 2

Limits: 0 to 28

Default: 0
```

Make Boot and OS Images Available on Boot Servers

Use an Image Prepare System on a Boot Server System

It is possible to install an IPS on a remote boot server and use the createosimage command to create a OS image locally. This is because IPS and the boot server share the image library. If the OS image has the same name as an already registered OS image, the OS image is accepted as present on the boot server.

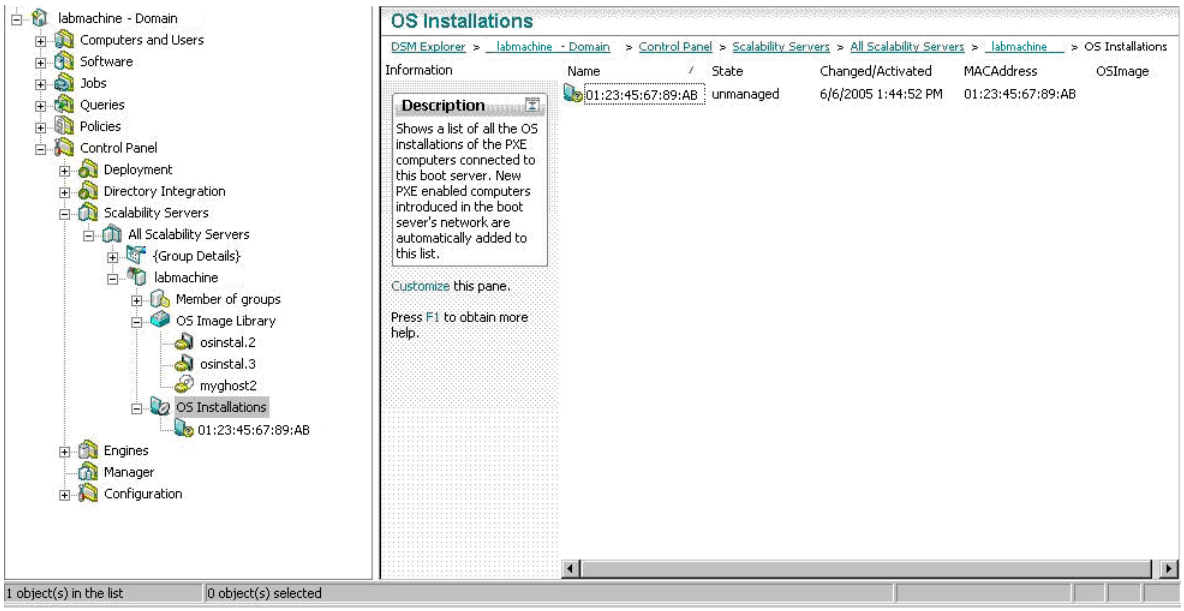
To create local boot images, use createbtimages.

Distribute Registered Images Using a Software Delivery Job

Registerosimage and registerbtimages can create a Software Delivery package of the OS / boot image in the Software Package Library. These images have an install procedure, “Add to boot server,” that makes the OS / boot image available in the image share of the destination boot server. Look in the Software Job | Job output tab for the installation state and log.

Boot Server View in the DSM Explorer

You can see the OS and boot images available in the image library of the boot server and the targets that the boot server is responsible for in the DSM Explorer under the Control Panel node, as shown in the following screenshot:



Boot Server on Windows Domain Controller

A Boot Server offering Share Access regularly changes the password of the user account canonprv. As canonprv refers to the same user account on all Domain Controllers inside a domain, changing the password on a Domain Controller would affect the share access of OSIM target machines to all other Boot Servers running on Domain Controllers in the same domain.

Therefore the automatic password update will be disabled when a Boot Server is installed on a Windows Domain Controller.

For security reasons the password of canonprv should be changed regularly when using share access. You can achieve this in one of two ways depending on your network environment:

- If you are running a Boot Server on only one Domain Controller inside a domain, you can enable automatic password changing by applying a configuration policy to the Boot Server with "Password change at Windows DC" set to True (1). The parameter is located under the DSM, Scalability Server, osim, ManagedPC,Server node.

- If you are running a Boot Server on more than one Domain Controller inside the same domain, you can configure the Boot Servers to use a password that has been manually set by the user.

  1. Create a separate configuration policy for each domain.
  2. Go to parameter section under the DSM, Scalability Server, osim, ManagedPC, Server node.
  3. Change the setting of the parameter "Canonprv's password" to "Centrally Managed" via its context menu.
  4. Set the value of the parameter "Canonprv's password" to the password you set for the domain account canonprv.
  5. Set the value of the parameter "Password change interval" to 0.
  6. Seal the policy and apply it to all Boot Servers in the domain.

  Note: The policy must be updated each time you manually update canonprv's password.

Note: When using DOS-based boot images to access shares at a Windows 2003 Active Directory domain controller, digital SMB signing must be disabled.

Open the Default Domain Controller Security Settings and set ‘Local Policies/Security Options/Microsoft network server: Digitally sign communications (always)’ to ‘Disabled’.

Use Boot Servers with or without MS Shares

The setup of the scalability server provides a boot server property sheet where the user can enable share access. Shares are not created if share access is disabled.

The boot server setup installs a switch tool, “sdbsswitch,” that can be used to change the boot server access method later.

The sdbsswitch tool creates or removes the OSIM shares and adjusts the OS images in the image store according to the access method:

- sdbsswitch -t switches from share to tftp access.
- sdbsswitch -s switches from tftp to share access.
- sdbsswitch -l shows the current configuration of the boot server.

Note: All OSIM images (except Ghost and ImageX images) are prepared to work with both share and tftp access.

Note: OSIM LINUX images always use NFS shares. The NFS share must contain the LINUX CD image. For more information, see "Installing LINUX from a Central NFS Server". The sdbsswitch command does not remove or create NFS shares.

Summary

- Installing an OS on a target is faster from a share because the setup installs directly from the share.
- OSIM shares are read-only for a special OSIM user. If your enterprise policy does not allow shares, the tftp download method is required.
- OSIM has implemented a special extended tftp protocol controlled by the OSIM boot server. In the case of tftp, all needed data are downloaded to the target before the installation starts.
- Unless the boot server and the Image Prepare System (see page 71) are not on the same system, sdbsswitch -t creates one <image>.caz file for every image and removes the directory tree.

  sdbsswitch -s will create the directory tree and remove <image>.caz.

sdbsswitch

The command sdbsswitch enables the boot server share or TFTP access.

Syntax

```
sdbsswitch -OPTION
```

OPTION

Valid values are as follows:

-L

Displays the current boot server configuration.

-S

Enables share access for the boot server.

-T

Enables TFTP access for the boot server.

Multiple Boot Servers in one IP Subnetwork

- You can install multiple boot server in the same IP subnetwork.
- The new boot server does not take responsibility for target computers already assigned to another boot server in the subnetwork.
- Each boot server is responsible for a specified set of target computers only. The DSM Manager determines the set of target computers.
- All boot servers observe PXE requests in their subnetwork, but they immediately reply only to requests of machines belonging to their own set of target computers. Requests from other machines are answered only after a certain number of retries and after a certain amount of time. The number of retries and the time to wait can be configured. More than one boot server may send such a default response.
- A target machine receiving one or more default responses will choose one of the replying boot servers and establish a connection. This usually is the boot server that replied first.
- A new target (not yet assigned to a boot server) will be fetched by that boot server.
- A moved target already assigned to a different boot server will be reassigned to the new boot server automatically. If there is an activated or pending installation order for the target and the required OS image is not available at the new boot server, the installation order is cancelled.

Configure the Boot Server and the DHCP Server to Co-exist on the Same System

Both the DHCP server and the boot server listen at the UDP port 67. If you have installed the DHCP server and the boot server on the same machine, ensure to either disable the boot server at the time of installing the manager or scalability server, or configure the boot server to stop listening at port 67. You must also configure the DHCP server so that a booting PXE client can be notified that there is a listening boot server on the network.

To configure the boot server to stop listening at port 67

1. Create a configuration policy for the boot server, if one does not exist, and modify the following policy settings under DSM, Scalability Server, OSIM, ManagedPC, Server:
   - Set "Enable DHCP proxy" to "False (0)"
   - Set "Use answer control list" to 0
2. Apply the policy to the boot server that co-exists with the DHCP server.

To configure the DHCP server

Configure the DHCP server by adding option 60 (class identifier) to the responses that the DHCP server sends to PXE clients. Depending on the operating system on which DHCP server is installed, perform the following steps:

Windows

1. Open DHCP Configuration and Management, right-click the server, and select Set Predefined Options.
2. Add option 60 to the defined options list with the value as "PXEClient".

Linux

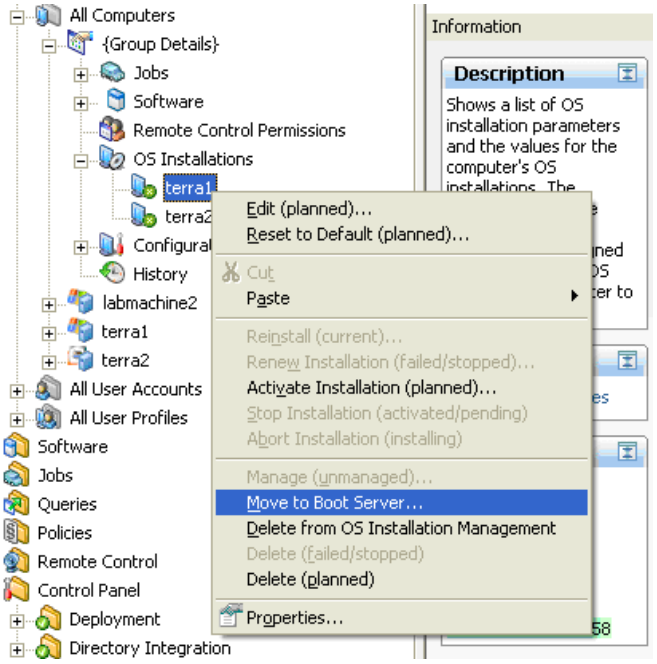
1. Set the value of "vendor-class-identifier" option to "PXEClient"

As described in the Preboot Execution Environment (PXE) Specification Version 2.1 (1999 by Intel Corporation), the PXE clients will now contact the boot server on the DHCP server's system using UDP port 4011 to get boot instructions. This enables the OSIM Boot Server to serve the PXE clients. There is no dynamic OSIM boot server assignment in this case. The OSIM boot server will not serve PXE clients that have been served by a different DHCP server, unless forwarded to the Boot Server using option 43.

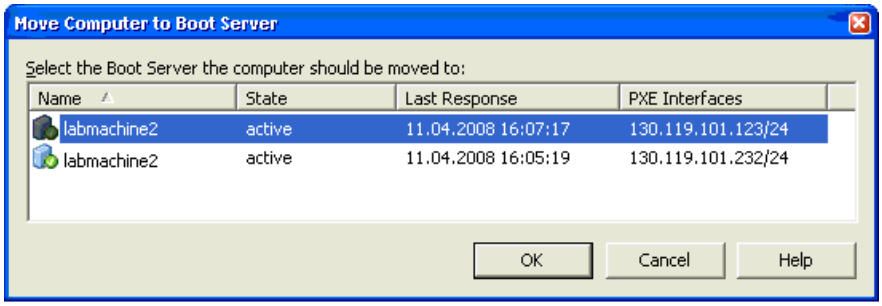
Note: With PXE 2.x clients, it is still possible to assign a different boot server via DHCP option 43 (encapsulated vendor-specific options) as described in Tech Document TEC381737.

Move an OSIM- managed Computer to a Boot Server Manually Using the Explorer

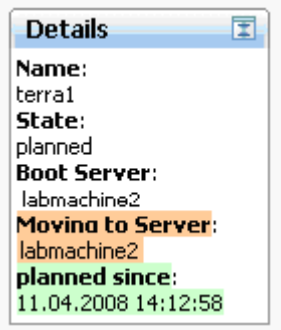
This method enables the user to attach an OSIM managed computer to a boot server. The function can be used to attach a boot server for the first time as well as to move the computer to another boot server as shown in the following screenshot:



Afterwards a dialog displays a list of all available boot servers of the domain together with the some boot server properties. The current attached boot server is selected initially as shown in the following screenshot:



After selecting another boot server and pressing the OK button, the computer move is in progress as stated in the computer's Details portlet below:



The move process normally takes only a few seconds. Hence it should be finished when the computer properties are reloaded next time as a result of a refresh.

It is also possible to select multiple computers to move them all together to a boot server.

Configure Boot Server Behavior with Common Configuration

The following configuration policy parameters under the DSM, ScalabilityServer, OSIM, ManagedPC, Server node change the behavior of a boot server with respect to other boot servers in the IP subnetwork:

Use ACL

Indicates whether the boot server uses an Answer Control List (ACL). Valid values are:

0

If Use ACL is set to "0", the boot server must be the only boot server in its IP subnetwork because it answers all PXE requests immediately.

1

If Use ACL is set to "1", the boot server answers PXE requests of assigned targets only, that is, the boot server uses an ACL.

PXE requests from other target machines will be answered only after a certain number of retries has been seen and after a certain amount of time has elapsed since the first request.

2

If Use ACL is set to "2", the boot server answers PXE requests of assigned targets only, that is, the boot server uses an ACL, but it does not answer to unknown targets.

Default: "1"

Discovery retries before answer

Number of retries before a boot server sends a default reply to the PXE request of a target not assigned to it.

Limits: “1” to “4”

Default: “3”

Note: Only evaluated if "Use ACL" is “1”.

Discovery timeout before answer

Number of seconds to wait before a boot server sends a default reply to the PXE request of a target not assigned to it.

Limits: “3” to “56”

Default: “10”

Note: Only evaluated if "Use ACL" is “1”.

The following table shows corresponding DiscoveryRetriesBeforeAnswer and DiscoveryTimeoutBeforeAnswer values:

DiscoveryRetriesBeforeAnswer	DiscoveryTimeoutBeforeAnswer
1	2
2	8
3	20
4	40

Any of the boot servers in an IP subnetwork can take responsibility for a PXE target. Therefore, it is recommended you store the needed OS and boot images on any of the boot servers in the subnetwork.

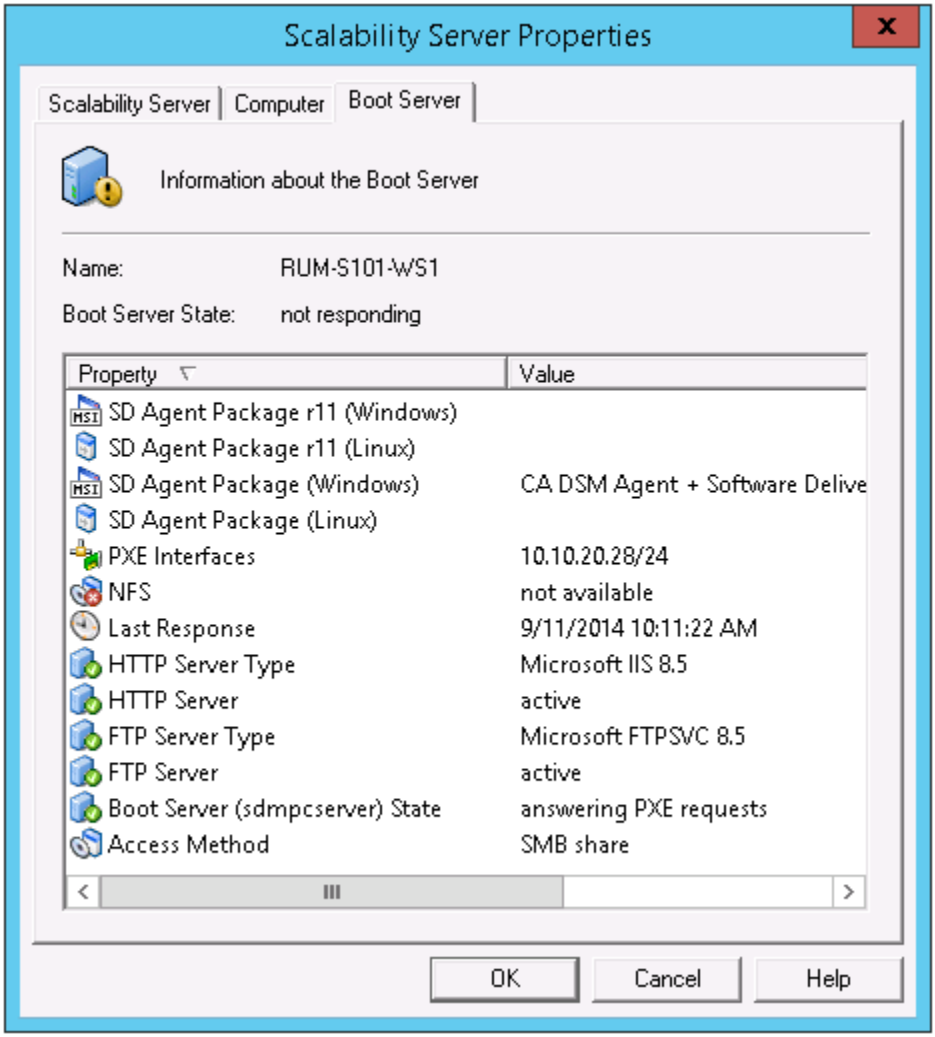
Linux, ESXi and Citrix Images Deployment and Staging

For Windows Boot Server, deployment and staging of the Linux, ESXi, and Citrix images is supported only with Server Class Operating System such as Windows Server 2008, 2008 R2, 2012, and 2012 R2.

Thus Boot Server installed on Desktop OS such as Windows7, Windows 8, and Windows 8.1 does not support OSIM deployment of Linux, VMware, and Citrix OS images. This is because of the unavailability of the NFS server on Windows Desktop OS. For more information, see Microsoft Windows Documentation.

Extended Boot Server Property Page

The boot server property page displays various details about the boot server:



The screen shot above displays all known boot server properties. Any GUI-unknown properties will be displayed as reported by the boot server, for example, property name and value are not localized.

Boot Server State

The Boot Server State together with the displayed icon reflects the state of the “sdmpcserver” plug-in as well as the boot server accessibility within the network.

active

The boot server is accessible in the network and answering PXE request.

failed

A boot server error is reported.

not responding

The boot server is not accessible in the network.

not running (stopped)

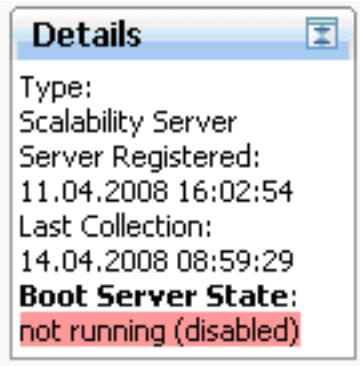
The sdmpcserver plug-in on boot server is stopped,

not running (disabled)

The sdmpcserver plug-in on the boot server is disabled.

Boot Server State Shown in Scalability Server Portlet

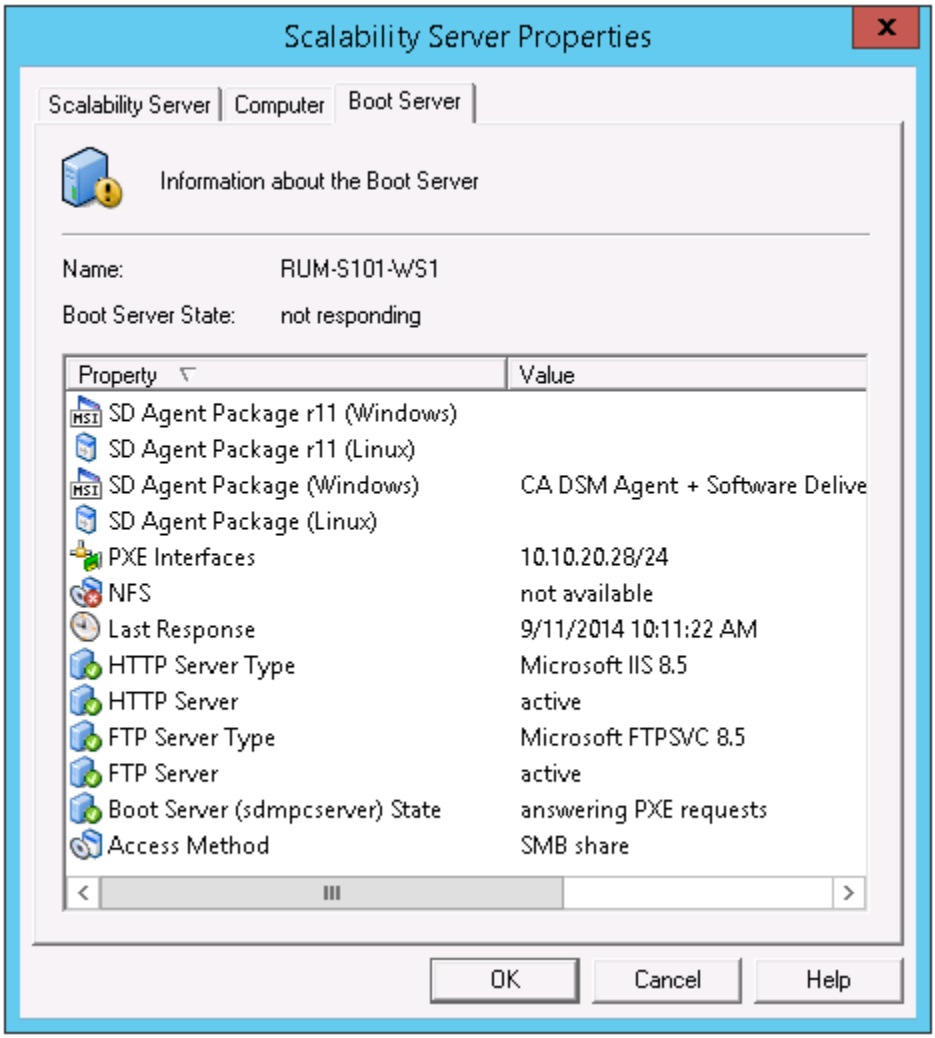
Any Boot Server State except “active” is also displayed in the Scalability Server portlet:



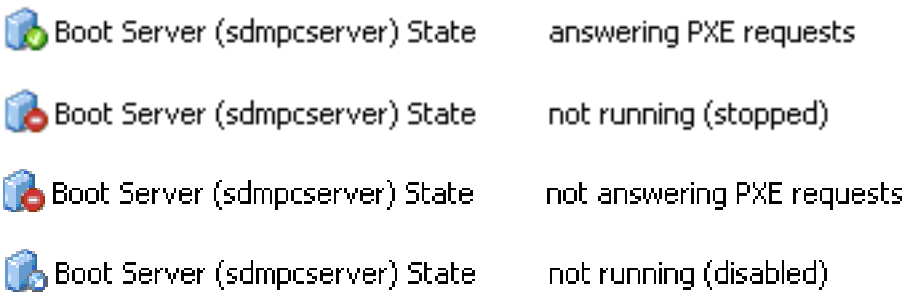
This portlet also shows details about a possible boot server error:

Boot Server Property Tab Error Details

On the property page, error details are displayed below the property list:



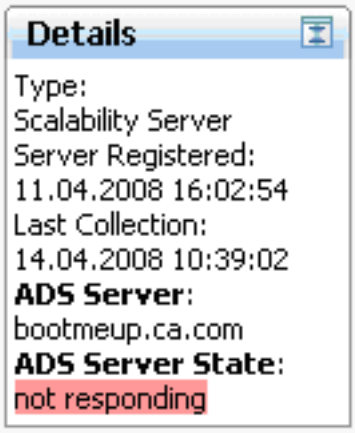
Boot Server (sdmpcserver) State



ADS Server State

ADS Server State	not responding
ADS Server State	active

The ADS server name and the ADS Server State, except “active”, are also displayed in the Scalability Server portlet:



Boot Server Access Method

Access Method	TFTP
Access Method	SMB share

NFS

NFS	not available
NFS	not active
NFS	active

Role

Role	Primary Domain Controller
Role	Backup Domain Controller

Using Microsoft ADS

This section contains the following topics:

OSIM Boot Server and Microsoft ADS Controller

PXE targets can alternatively be managed by an ADS (Automated Deployment Services) server.

Note: The Microsoft ADS Controller must not be installed on the OSIM Boot Server or on any DHCP server that is replying to OSIM PXE target machines; otherwise one of the boot servers may not be able to serve PXE targets.

If the ADS server is configured in the OSIM boot server, the OSIM boot server will ask the ADS server before it answers PXE requests of a special target. If the ADS server is responsible for the target, the OSIM boot server will not reply to PXE requests from the target.

Specifically, if no connection to the ADS server can be established, the configured OSIM boot server will only reply to PXE requests from target computers assigned to the OSIM boot server and not yet ADS managed.

Otherwise, if the target computer is already ADS managed, the configured OSIM boot server will never reply; and if the target computer is not ADS managed, it will follow the useACL (see page 59) settings.

Note: The precise meaning of "ADS managed" is determined by the ADSDload settings (see page 59).

If the customer adds PXE targets to the ADS server, the configured OSIM boot server will be notified from the ADS server. As the boot server reports the ADS managed targets to the DSM domain manager, this targets will be marked as ADS managed in the MDB database. The domain manager removes all OS installation jobs of that target and also the current OS installation will be removed. The domain manager does not provide methods to create OSIM OS installation jobs for the target as long as the target is marked as ADS managed.

If the customer removes the PXE target from the ADS server, the configured OSIM boot server will be notified from the ADS server. The boot server reports the target as no longer ADS managed to the domain manager.

You have to use the OSIM tab of the Computer Properties dialog to enable the computer for OSIM again.

Note: ADS must be configured to use MAC addresses as device identifiers (this is the default).

Automated Deployment Services (ADS)

Microsoft Automated Deployment Services (ADS) is a deployment framework used to remotely and automatically deploy operating systems in the Microsoft® Windows® 2000 Server and Microsoft Windows Server 2003 families, as well as to configure, maintain, and manage servers.

ADS provides a remote deployment infrastructure that consists of tools and technologies that can install an operating system on a "bare metal" server or re-provision an existing server with a different OS image.

The following are high level tasks that ADS can perform:

- Prepare the server to be in the ADS environment.
- Capture an image of the operating system from the existing system.
- Deploy the image to a target system.

Configure the ADS Server in the OSIM Boot Server

When ADS has been installed and set up on a Windows 2003 server, you need to enable the ADS connection for the scalability server by changing the configuration of the OSIM, ManagedPC, Server policies.

Define and Distribute Configuration

Define a configuration policy on the domain manager of the scalability server and distribute the configuration using a configuration job. The ADS-related configuration policies are available under the DSM, Scalability Server, OSIM, ManagedPC, Server node.

Note: For more information about configuration policies, see the Configuration Policy section of the *DSM Explorer Help*.

Establish Boot Server - ADS Communication

The following parameters must be set to allow the boot server to communicate with ADS:

ADSUse

Must be set to 1, or true.

ADSProvider

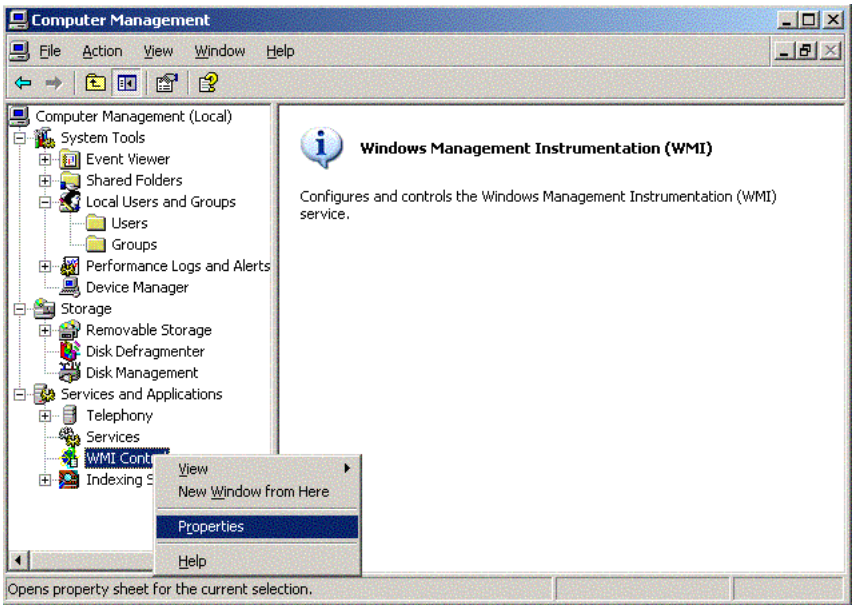
The IP address or host name of the machine that contains the MicrosoftADS namespace.

ADSUserId

The ID of the user, who must be a member of the Administrators group on the ADS Controller and must have been granted full permissions to the \\ADSProvider\ROOT\MicrosftADS namespace.

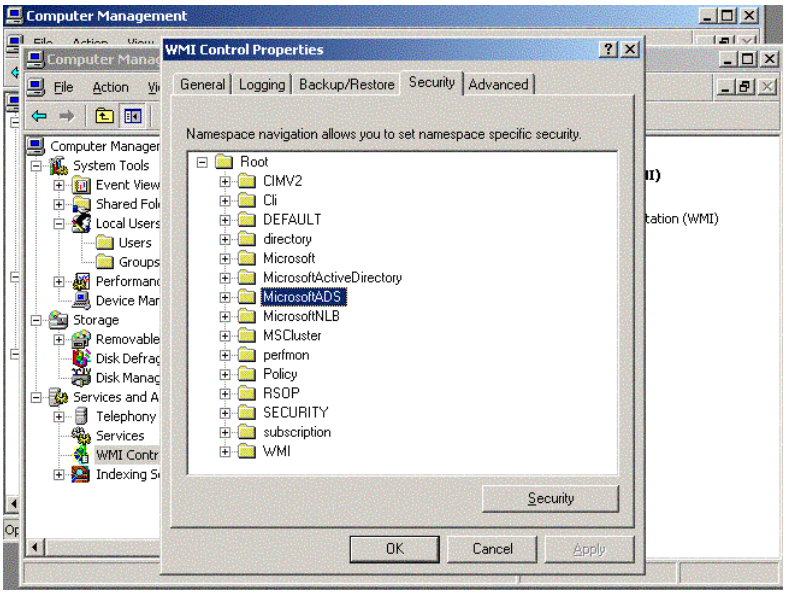
If the user is not the “Administrator,” then a new user's access rights can be set by following these procedures:

1. First create a new user from Computer Management, and then configure WMI by going to the WMI Control Properties dialog as shown in the following screenshot:

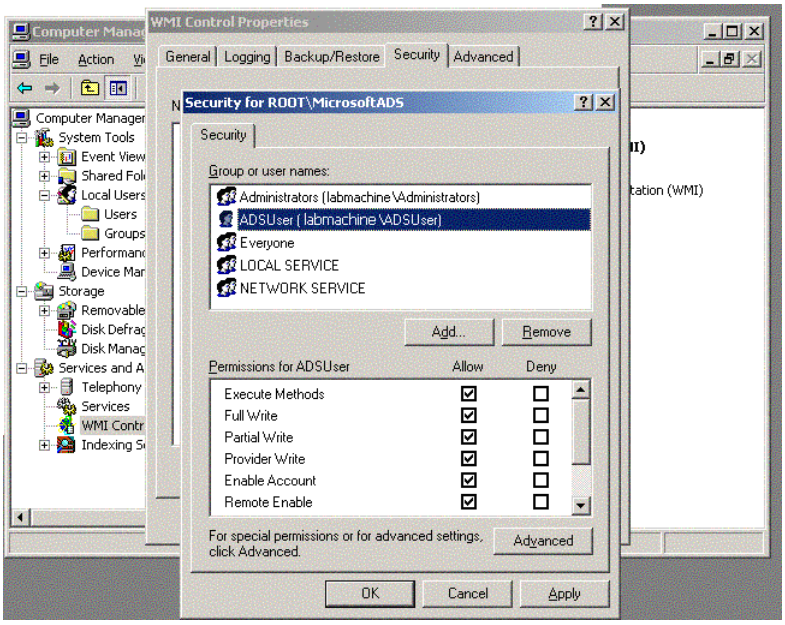


2. Select the MicrosoftADS namespace and click the Security tab.

   Thw WMI Control Properties dialog appears as shown in the following screenshot:



3. Then add the following rights for the new user as shown in the following screenshot, and click OK:



ADSPassword

The password to log in to the ADS controller for the given ID.

ADSDomain

The domain to use to authenticate the user when connecting to the controller. This can be left blank if the user to be authenticated has been defined on the ADS controller.

ADSAuthenticationType

The type of authentication to use when connecting to the ADS controller. This value can be set to either ntlmdomain or Kerberos

Configure Microsoft ADS

To configure Microsoft ADS

1. Follow the ADS documentation on managing devices to be controlled by the ADS controller.
2. Whenever a device is modified in the ADS database, the scalability server will receive notification of this and report the changes to the OSIM manager.
3. If devices already exist in the ADS database, these devices will not automatically be visible to the OSIM manager until they make a PXE request.

Chapter 4: Using the Image Prepare System

Performing an installation of an operating system requires creating images of the system you want to install, an OS image and boot images. The OS Installation environment offers an Image Prepare System (IPS) that lets you build such images with specific commands.

You can install the IPS separately from the OS Installation Manager or on the same system as the OS Installation Manager.

The IPS provides commands that introduce the operating system and boot images to the OS Installation Manager and to Client Automation. The only way to introduce OS images in OSIM is by using these commands.

The IPS commands are available at the GUI of the DSM Explorer where Wizards guide you step by step through the creation and registration of boot and OS images. And you can use the commands directly at the DOS Command Prompt and integrate them into your own programs.

This section is the reference of the command line interface of the IPS commands. At the GUI, all necessary information is given within the Wizards.

Note: You must have the administrator privileges to run the IPS commands at the command line interface.

This section contains the following topics:

CreateBTImages--Manage Boot Images

Valid on Windows

The Image Prepare System command createBTImages provides the following functionality:

- Creates new boot images

  `createBTImages -i imagename ...`

- Lists the boot images available at the Image Prepare System

  `createBTImages -x`

- Deletes the specified boot images

  `createBTImages -d imagename`

If you do not set the -x or -d flags, the createBTImages command creates a new boot image.

The command has the following format:

```
CreateBTImages
    {-x | -d <imagename> | -i <imagename> -o <type> | -i <imagename> -e}
    [-s <winpeimgpath>]
    [-c <winpepath>]
    [-w <ospath>]
    [-t <driverpath>]
    [-n <loader>]
    [-b <floppydrv>]
    [-l <lanman>]
```

where:

-x

List of all created boot images

-d<imagename>

Deletes boot images (<imagename>[<extension>])

-i<imagename>

Creates boot images (<imagename>[<extension>]).

Without ext it creates for DOS: .2 (disk partition image)

and .3 (OS install image). It creates for WinPE, WinPE2, WinPE3 only .2

Default: osinstal.2 and osinstal.3 or winpe.2, winpe2.2, winpe3.2, winpex64.2, winpe264.2, winpe364.2

-o<type>

Specifies the Boot Image platform.

Possible platforms are:

- DOS = MS-DOS 6.22, Windows 95, Windows 98

  Note: DOS boot images are not FIPS-compliant; hence cannot be used in FIPS-only mode.

- DOSx = MS-DOS 6.22, Windows 95, Windows 98 for LINUX

  Note: DOSX boot images are FIPS-compliant but are supported only for LINUX deployments.

- WINPE = Windows PE 1.x
- WINPEx64 = Windows PE 1.x x64
- WINPE20 = Windows PE 2.0
- WINPE20x64 = Windows PE 2.0 x64
- WINPE30 = Windows PE 3.0
- WINPE30x64 = Windows PE 3.0 x64
- WINPE40 = Windows PE 4.0
- WINPE40x64 = Windows PE 4.0 x64
- LINUXPEx64 = Linux-based customized CentOS 6.3 x64 boot image

-e

Deletes an existing boot image and creates a new image with the same name and type.

For WinPE20, WinPE30, WinPE20x64, and WinPE30x64:

```
CreateBTImages –i <image name> -e
```

For WinPE and WinPEx64:

```
CreateBTImages –i <image name> -c <winpepath> [-w <ospath>] -e
```

image name

Specifies the name of an existing image that you want to delete and recreate.

winpepath

Specifies the path to Windows PE 2005.

-s<winpeimgpath>

Specifies the temporary directory to prepare WinPE image files

(winpe.iso, winpe.wim)

Default: Temporary directory in <image store>\bttemp\<type>\

-c<winpepath>

Specifies the source path of the Windows PE directory or the Windows AIK depending on the version of Windows PE.

- **WinPE2005:** Path to the WinPE directory. The WinPE directory must contain a copy of the WinPE directory from the CD, "Microsoft Windows Preinstallation Environment 2005" (Microsoft Part No. T94-00005).
- **WinPE 2.0 and WinPE 3.0**: Install path of the Windows AIK 1.1 and 2.0 respectively. Download the Windows AIK (WAIK) DVD from the Microsoft web site and install Windows AIK from the DVD. By default, the install directory is c:\program files\Windows AIK.

-w<ospath>

Specifies the path to Windows OS files to create a WinPE 1.x image

Default: Path to Windows OS (Windows 2003 Server SP1 for 32-bit, Windows 2003 Server x64 or Windows XP x64 for 64-bit) files on CD

This option is not required for WinPE20, WinPE20, WINPE20x64, or WINPE30x64

-t<driverpath>

Specifies the path to additional drivers (*.inf, *.sys) for a WinPE image. For WinPE20, WinPE30, WinPE20x64, and WinPE30x64, you can store the drivers in subdirectories under the given driver path. Driver files are added to the driver directory of the boot image template in the image prepare system. The drivers are then inserted in any new boot image based on the type and template files. By default, this switch adds the drivers that already exist in the boot image template.

-n<loader>

Specifies the name of the WinPE network loader.

Default: startrom,pxeboot.n12

-b<floppydrv>

Specifies the floppy disk drive with DOS floppy.

Default: First drive of a bootable floppy disk, for example, a:

-l<lanman>

Specifies the path to Microsoft LAN Manager files for DOS; these files are available on the Microsoft Server or Windows NT Server 4.0 CD

Default: Path to the LAN Manager files on Windows NT Server 4.0 CD

Notes: (for DOS and DosX boot image)

Provide a bootable, write-enabled floppy disk based on MS-DOS 6.22, Windows 95 or Windows 98, when you create boot images. For more information, see Creating DOS Boot Images.

If you create boot images the first time after installing the Image Prepare System, you also have to:

- copy 'sys.com', 'format.com', 'smartdrv.exe' to the bootable floppy disk.
- (DOS) Provide the LAN Manager client with –l <path> or insert the Windows NT Server 4.0 CD into your CD drive.

CreateOSImage--Managing OS images

Valid on Windows

The Image Prepare System command createOSImage provides the following functionality:

- Create a new OS image:

  `createOSImage -i imagename -o OStype ...`

- List all the OS images available at the Image Prepare System:

  `createOSImage -x`

- Delete a specified OS image from the Image Prepare System:

  `createOSImage -d imagename`

- Show detailed information about a specified OS image:

  `createOSImage -p imagename`

- Create a .caz file of the image directory, in order to test the image with TFTP download:

  `createOSImage -z imagename`

This command has the following format:

```
CreateOSImage {-i<imagename> -o<OStype> | -x | -d<imagename> | -p<imagename> |
-z<imagename> | -i<imagename> -e}
[-s <OSpath>] [-t <driverpath>] [-l <locale>] [-k <Prod.Key>] [-r <resp. file>] [-a
<image>] [-g <imgtools>] [-n <path to language packs>] [-h <path to hotfixes>]
```

-i<*imagename*>

Creates an OS Image.The image name can consists of [a-z, A-Z, 0-9, -, _] with a maximum of 32 characters.

-x

Lists all created OS images.

-o<ostype>

Specifies the operating system type of the image. For more information about the supported OS and their types, see the Supported OS (see page 13) section.

-d<*imagename*>

Deletes the OS image.

-p<*imagename*>

Lists the information of the OS image.

-z<*imagename*>

Creates one image file for non-share download additionally to the tree.

-e

Updates the specified image with the current templates. For more information, see Updating OS Images (see page 19).

-s<OSpath>

Specifies the path to the operating system files such as winnt.exe, setup.exe, and so on.

Default: CD drive

-t<*driverpath*>

Specifies a path to additional drivers (for example, *.inf, *.sys). The drivers are added to the OSIM OS image. They are recognized during the hardware search of the OS installation on the target. This option cannot be used with the -a option that specifies an existing OS image. Instead, you must add the drivers to the existing OS image.

You can use the -t option with the -e option to update the drivers of an existing OS image:

```
createOSImage –i <imagename> -e –t <driverpath>
```

You can create an SD update package that includes the updated drivers using the following command:

```
registerOSImage –i <imagename> -e –s <manager> -l
```

-l<*locale*>

Specifies the code page used to map the boot parameter during OS setup

Default: ENU Latin

- THA = Thai, JPN = Japanese, CHS = Chinese (PRC, Singapore),
- KOR = Korean, CHT = Chinese (Taiwan, Hong Kong SAR, PRC),
- ENU = Latin (US, Western Europe), HEB = Hebrew,
- ROM = East European (Romania, Hungary, Poland, Czech Rep ...)

-k<*Prod.Key*>

Specifies the product key of the OS.

Default: 000-0000000

-r<*resp. file*>

Specifies the path to an own responsefile that should be used.

Default: Image Prepare System provided template

-a<*image*>

For Windows, it specifies an existing image which provides the image files for installation. Such an image has only its own control files in CAMENU.

For LINUX, –a gives an external NFS share name including the LINUX packages normally on a central NFS server. The created LINUX image includes the setup files only. See chapter LINUX OS images (see page 181).

-g**<*imgtools*>**

Applicable only for GETIMAGE image type

Specifies the path to the imaging tools such as imagex.exe and ghost32.exe that are used to capture a model computer. You can specify multiple paths, separated by a semicolon.

Note: You need not specify the path to imagex.exe if Microsoft Windows AIK is installed on the Image Prepare System.

-n<path to language packs>

Applicable only for Windows 7 and higher versions

Specifies the path to language packs. Microsoft provides language packs on separate DVDs. The DVDs contain subdirectories for every language. Use this parameter to prepare an OSIM Windows image to install multiple languages. You can specify the language of the target OS installation in the OS installation job. The language packs, in the path specified with this parameter, are copied to the OSIM OS image store and is used to install the correct language at the target computer. The language of the target OS depends on the OS installation job parameter setting and the available language pack; The fall back language is ENU.

Note: Alternatively, you can manually copy the respective language packs from DVDs to the Windows OS image in the image store at <OSIM-image store>\images\<imagename>\<imagename>\oeminst\langpacks.

-h<path to hotfixes>

Applicable only for Windows images

Specifies the path to OS hotfixes. The hotfixes, in the path specified with this parameter, are copied to the OSIM OS image store and is applied on the target computer as a post-installation procedure.

Note: Alternatively, you can manually copy the hotfixes to the OS image in the image store at <OSIM-image store>\images\<imagename>\<imagename>\oeminst\hotfixes.

RegisterBTImages--Registering Boot Images

Valid on Windows

The Image Prepare System command registerBTImages provides the following functionality:

- registerBTImages -s *servername*

  Registers a boot image at a server.

- registerBTImages -s *servername* -b

  Registers a boot image only at the OS Installation Manager.

- registerBTImages -s *servername -l*

  Registers a boot image SD package only at the domain manager.

- RegisterBTImages -w *directory*

  Registers from a specified directory including an exported boot image SD package.

Note: RegisterBTImages cannot be started more than once concurrently on the same IPS.

The command has the following format:

```
RegisterBTImages {-s <manager>} [-i <imagenames> | -w <directory> ] [-b] [-l] [-e]
[-m <dosfiles>] [-n <name>] [-v <version>] [-c <comment>] [-r] [-u <user> -p <password>
-d <domain>]
```

-s<*manager*>

Specifies the name of the DSM-Manager to register the boot images

-i<*imagenames*>

List of the boot image files, separated by semicolon

Default: osinstal.2;osinstal.3

-w<*directory*>

Register from directory including the exported Boot image-SD package

-b

Register only parameter definitions of the BOOT images

-l

Register only the Boot images in the SD software library. You can register a Boot image without specifying the -b or -l option. In this case, the image's parameter definitions are registered before the Boot image is registered in the SD software library. If the SD registration fails or is aborted, the image's parameter registration will still be in effect. You just need to specify the -l option to register the image again.

-m<*dosfiles*>

List of additional needed files, separated by semicolon

Default: format.com;smartdrv.exe

-e

Registers a boot image update package. The default SD package name is extended with –upgrade.

-n<*name*>

SD product name of the Boot images in the Software Library

Default: Name of the Boot Image

-v<*version*>

Specifies the SD product version of the Boot images

Default: 12.0/00

-c<*comment*>

Specifies the comment that appears in the SD product

Default: BsmBtImg <name>

-r

Specifies a complete DOS boot disk. No additional files are required

-u<*user*>

Specifies the remote DSM-Manager user

-p<*password*>

Specifies the password of the user

-d<*domain*>

Specifies the remote DSM-Manager domain, for example:

```
winnt://manager
```

RegisterOSImage--Registering OS images

Valid on Windows

The Image Prepare System command registerOSImage provides the following functionality:

- registerOSImage -s *managername* -i *imagename*

  Registers an OS image from the Image Prepare System with the addressed OS Installation Manager and its associated domain manager.

- registerOSImage -s *managername* -i *imagename* -b

  Registers only parameter definitions of the image with the addressed manager.

- registerOSImage -s *managername* -i *imagename* -l

  Registers only the operating system to the addressed manager.

- registerOSImage -i *imagename* -t

  Checks an OS image for correct OS Installation Manager parameter usage.

- registerOSImage -w *directory*

  Registers from the specified directory including the exported OS SD package.

Note: Though RegisterOSImage can be started more than once on the same IPS, it does not allow the registration of the same OS Image to different domain managers concurrently.

The command has the following format:

```
RegisterOSImage -s <manager> -i <imagename> | -w <directory> [-b] [-l] [-t] [-e] [-n
<name>]
    [-v <version>] [-c <comment>] [-u <user> -p <password> -d <domain>]
```

where:

-s<*manager*>

Specifies the name of the DSM-Manager to register the OS image

-i<*imagename*>

Specifies the name of OS image directory (see CreateOSImage)

-w<*directory*>

Register from directory including the exported OS-SD package

-b

Register only parameter definitions of the OS image

-l

Register only the OS image in the Software Package library. You can register an OS image without specifying the -b or -l option. In this case, the image's parameter definitions are registered before the OS image is registered in the SD software library. If the SD registration fails or is aborted, the image's parameter registration will still be in effect. You just need to specify the -l option to register the image again.

-t

Only test the OS image parameter description

-e

Registers an OS image update package. The default SD package name is extended with –upgrade.

-n<*name*>

Specifies the SD product name of the OS image in the Software Library

Default: <*imagename*>

-v<*version*>

Specifies the SD product version of the OS image

Default: 12.0/0

-c<*comment*>

Specifies the comment that appears in the SD product

Default: BsmOsImg <name>

-u<*user*>

Specifies the remote DSM-Manager user

-p<*password*>

Specifies the password of the user

-d<*domain*>

Specifies the remote DSM-Manager domain, for example,

```
winnt://manager
```

OS image Detailed Information

OS and boot images have their own description file, osinfo.ini that contains detailed information about the images. The following sections describe the contents of the description file for OS and boot images.

OS Image

An OS image description file osinfo.ini is stored in the OSIM OS image store along with the image. You can either open the file or view its contents using the following command:

```
createosimage -p <imagename>
```

The osinfo.ini file contains the following sections:

- images section (see page 83)
- parameter section (see page 85)
- camenu section (see page 86)
- parameterfile section (see page 87)
- Update section

[images]

The images section presents mainly the following information:

- locale information (see page 83)
- imageflags (see page 84)
- imagename (see page 84)
- sharename (see page 85)
- usedshare (see page 85)
- ostype (see page 85)

locale=

Specifies the code page to use during the OS setup.

"Language identifiers (see page 103)" shows a list of valid keys.

Default: ENU

OSIM imageflags=

Defines the features or properties of the OS image. The value of this property can contain a comma-separated list of key words.

Note: The key words are case-sensitive.

Following tables provides the list of key words that you can use with this parameter:

Keyword	Definition
F140-2	Specifies that the image can accept parameters in FIPS-compliant encryption only
DosTftp	Specifies that the image supports installation using a DOS-based boot image in TFTP mode.
DosLm	Specifies that the image supports installation using a DOS-based boot image in share mode.
PeTftp	Specifies that the image supports installation using a WinPE-based boot image in TFTP mode.
PeLm	Specifies that the image supports installation using a WinPE-based boot image in share mode
Pe2Lm	Specifies that the image supports installation using a WinPE20-based or WinPE30-based boot image in share mode
Pe64Tftp	Specifies that the image supports installation using a WinPEx64-based boot image in TFTP mode.
Pe64Lm	Specifies that the image supports installation using a WinPEx64-based boot image in share mode
DosLmT	Specifies that the image supports Linux-based boot image using TFTP in share mode.

imagename=

Specifies the name of the OS image.

productid=

Specifies the product key of the operating system.

Default: 000-0000000

osplatform=

Specifies the name of the operating system.

imagetime=

Specifies the date and time of creation of the image.

sharename=

Specifies the name of the share where the image is stored. The share is located at MANAGEDPC\images relative to the installation directory of CA\DSM\server\SDBS\var.

usedshare=

If the OS image uses the image of another OS image, this entry refers to the share for this base image. It is located at MANAGEDPC\images. If the OS image uses an installation image of its own, the entry carries the same value as the entry sharename.

ostype=

Identifies the type of operating system associated with the OS image.

For valid values, see usage of CreateOSImage.

Sdotype=<number>

Alternatively, a list of numbers separated by commas.

List of Software delivery OS types to determine compatibility with Software Delivery distribution packages.

Createshare=MS, MSNFS

Specifies what type of shares the images uses. MS means Microsoft shares, MSNFS means Microsoft shares and NFS shares.

createzip=<name of the zip file>

If createzip is set with a name of a zip file, the OS image is prepared to install from a TFTP boot server. In the case of a TFTP boot server, the zip file is downloaded to the target instead of installing from a Microsoft share. Because Linux CD and DVD images are very large, Linux CD's are not transferred in a zip file to the target. The target accesses Linux CD's always from an NFS share.

[parameter]

The parameter section presents the following information:

default=

This parameter refers to the default file that describes the operating system parameters. This file is normally called default.ini and is located at the share specified by the sharename.

[camenu]

The camenu section specifies files that control the OS installation process of this OS image. The files are all located at MANAGEDPC\CAMENU relative to the installation directory of the Boot Server / IPS image store.

batfilename=

If the boot server is using shares, identifies the main installation script, or batch file, that starts from the DOS boot image and prepares and launches the installation.

tftpfilename=

If the boot server is using tftp, identifies the main installation script that starts from the DOS boot image and prepares and launches the installation.

responsefile=

Specifies the name of the auto-answer file that contains all the information needed for an unattended installation of the related operating system. This file is compliant with Microsoft's "Answer File Parameters for Unattended Installation of Windows (2003, 2000, XP, ME, 98)" or Linux unattended installation.

partition=

Specifies the file that defines the partitions of the hard disk.

[parameterfile]

Identifies the files that contain OS installation parameters. If you have defined the parameters in some other files, you must add the path to those files under this section.

When the RegisterOSImage command executes, it searches the given files for additional parameter definitions, adds the parameters found to the default.ini file, and registers them with the OS installation manager.

If a parameter already exists in the default.ini file, RegisterOSImage tries to register the parameter using the following command:

```
Registerosimage –i <image name> -t
```

The new parameters get a default definition in default.ini; you can however, modify them by editing the default.ini file. If you have edited the default.ini file, execute the following command to add or update the parameter definitions to the MDB:

```
Registerosimage –i <image name> -s <manager> -b  [-e]
```

file1=

Name of the first file containing OS Installation Manager parameters. The name is relative to the image store directory of the Boot Server or IPS.

The default location of the image store is \Program Files\CA\DSM\Server\SDBS\var\managedpc.

filex=

Name of the last file containing OS Installation Manager parameters. The name is relative to the MANAGEDPC directory of the Boot Server / IPS image store.

[update]

The update section lists the files that IPS must add or delete from the OS image to migrate the image to the new template. This information is used to create an update SD package for an OS image.

addfile<n>=

Specifies the name of a new file that IPS must add at the time of updating the OS image to a new image template. The path is relative to the image store directory of the Boot Server or IPS.

The default location of the image store is \Program Files\CA\DSM\Server\SDBS\var\managedpc.

Example: addfile parameter

addfile1=camenu\file.inf

addfile2=images\myimage\myimage\oeminst\driver\extdrv.sys

delfile<n>=

Specifies the name of the file that IPS must delete at the time of updating an OS image to a new image template.

[installed]

This section includes the information that product installations use while installing the OS image.

sdproductname<n>=<

Specifies the name of the product.

sdproductversion<n>

Specifies the version of the product.

Boot Image

The description file for a boot image is stored in the <image store>\images\dosboot\undi\<boot_image_name>.

The description file for a boot image contains a [BootImage] section with the following parameters:

Imageflags

Defines the features or properties of the boot image. The value of this property can contain a comma-separated list of key words.

Note: The key words are case-sensitive.

Following tables provides the list of key words that you can use with this parameter:

Keyword	Definition
F140-2	Specifies that the image can accept parameters in FIPS-compliant encryption only
DosTftp	Specifies that the image is DOS-based and supports installation through TFTP mode
DosLm	Specifies that the image is DOS-based and supports installation through share mode **Note:** This key word cannot be combined with the key word F140-2 as DOS images are not FIPS-compliant.
PeTftp	Specifies that the image is WinPE-based and supports installation through TFTP mode
PeLm	Specifies that the image is WinPE-based and supports installation through share mode
Pe2Lm	Specifies that the image is WinPE20-based or WinPE30-based and supports installation through share mode
Pe64Tftp	Specifies that the image is WinPEx64-based and supports installation through TFTP mode
Pe64Lm	Specifies that the image is WinPEx64-based and supports installation through share mode
DosLmT	Specifies that the image is DOS-based and uses TFTP in share mode. When Client Automation is operating in FIPS-only mode, use this key word for Linux OS installations that use DOS-based boot images

Boottype

Specifies the type of the boot image. Following are valid values:

- DOS = MS-DOS 6.22, Windows 95, Windows 98
- WINPE = Windows PE 1.x
- WINPEx64 = Windows PE 1.x x64
- WINPE20 = Windows PE 2.0
- WINPE20x64 = Windows PE 2.0 x64
- WINPE30 = Windows PE 3.0
- WINPE30x64 = Windows PE 3.0 x64
- WINPE40 = Windows PE 4.0
- WINPE40x64 = Windows PE 4.0 x64
- LINUXPEx64 = Linux-based customized CentOS 6.3 x64 boot image

For LinuxPEx64, See Create and Register a Boot Image using DSM Explorer (see page 151)

Bootimagedir

Specifies the directory that contains boot image files.

Bootloader

Specifies the network loader for the boot image, which is stored in the directory specified in Bootimagedir.

Switchfile

Specifies that the boot server must switch to the next boot image or boothd if the specified file is downloaded from Bootimagedir.

Sample winpe2.2 file:

```
[BootImage]

Imageflags
Boottype=WinPE2
Bootimagedir=winpe2.2
Bootloader=pxeboot.n12
Switchfile=winpe.wim
```

Default.ini

The default.ini describes the OS Installation Manager parameters used in an OS image and the default settings for these parameters. The file is a UTF-8 coded file and should be edited with UTF-8 enabled editors only e.g. Notepad. The file has a .ini file structure. The file shows the following sections:

- Locale Section
- Default Section
- Reserved Section
- Parameter Sections

locale

The code page 3 indicates that this file is a UTF-8 file.

default

This section contains the default settings for the parameters. The entries use the following format:

parameterName >= defaultValue

reserved

This section lists a number of names that should not be used as OS Installation Manager parameter names. The entries of the section use the following format:

name=parameter

Default.Ini Parameters

The default.ini parameter sections have the following structure:

[*parametername*] Type={ Text (see page 92) | Pwd (see page 92) | IP (see page 92) | Num (see page 93) | BootImage (see page 93) | OSImage (see page 94) | Mac (see page 94) | MapList (see page 94) | MapListExt (see page 94) | TextList (see page 95) | TextListExt (see page 95) }

The remaining entries of a parameter definition section and their ranges depend on the parameter type (Type=).

Text Parameter

Provides any simple text string. The length of the string can be limited by the MaxLength attribute. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the strings to be assigned	1 to 255 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

Password Parameter

Provides an encoded password. The password can be limited to MaxLength characters. A password is always related to a user ID. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the password to be assigned	1 to 255 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

IP Parameter

Specifies an IP address of the format a.b.c.d where each a...d represents an integer value between 0 and 255. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the IP address	17 characters
Comment	Comment on the parameter	any string

Attribute	Meaning	Range
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

Numeric Parameter

Specifies a numeric value. The range of the value is given by the attributes MaxValue, which specifies the maximum, and MinValue, which specifies the minimum. The value of MinValue must be less than or equal to the value of MaxValue. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxValue	Maximum of the value assigned	-2147483648 to 2147483647
MinValue	Minimum of the value assigned	-2147483648 to 2147483647 and MinValue <= MaxValue
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

Boot Image Parameter

Specifies the name of a boot image. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the boot image name	1 to 32 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

OS Image Parameter

Specifies the name of an operating system image. The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the operating system image name	1 to 10 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

Mac Parameter

Specifies a MAC address of the format ab.cd.ef.gh.ij.kl where each a...l represents a hexadecimal digit (0-9, a-f). The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of the IP address	12 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}

Mapping List / Mapping List Extended Parameter

Specifies a list of item=<string1> <string2>. String2 is shown in the editor only, when string1 has been assigned a value.

The parameter editor can extend parameters of type MapListExt, but not of type MapList.

When assigning a value to a parameter, it can only be a string that is appearing in the list.

The following parameter attributes are valid for these parameter types:

Attribute	Meaning	Range
MaxLength	Maximum length of a list entry	1 to 255 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}
Item	Specifies a list item. This attribute can occur more than once.	"string1" "string2"

Text List / Extended Text List Parameter

TextListExt specifies a list of item=<string>; that is, it is the same as a TextList parameter, but the list can be extended with the parameter editor later on.

The following parameter attributes are valid for this parameter type:

Attribute	Meaning	Range
MaxLength	Maximum length of a list entry	1 to 255 characters
Comment	Comment on the parameter	any string
Trans	Drives the inheritance of parameter settings	{yes|no}
UpdateDefault	Overwrites the default value when updating the image	{yes|no}
Item	Specifies a list item. This attribute can occur more than once.	any string

UpdateDefault

The RegisterOSImage command provides an option to overwrite the default values of the parameters in the database. For the overwrite operation to be successful, you must set the UpdateDefault attribute of the parameter to Yes; Otherwise, the default value are not overwritten. The following RegisterOS command overwrites the default values:

```
RegisterOSImage –i <image_name> -s <manager> -e [-b]
```

Trans Attribute

Drives the parameters settings when an operating system image is assigned to a target. When an operating system image has a predecessor image (Current or Scheduled configuration) for the target in question, a parameter of the new image also occurs in the old image, and both parameters are of the same type, then the parameter of the new image inherits the parameter setting from the predecessor configuration provided that the Trans attribute of the parameter at the new image has been set to yes. In any other case, the parameter of the new image is set to its default, if there is any, or it remains unset.

Example: Add OSIM OS Image Parameters (see page 26)

Explanation of [localeID] in default.ini:

The parameter created is an Extendable Map List (Type=MapListExt). The item values have a maximum length of 128 bytes (MaxLength=128). When the OSIM OS image is assigned to a target computer, the value of this parameter can be inherited from the scheduled configuration or the current configuration (Trans=yes).

The example contains entries for Chinese Macau, Danish, English US, English UK, and French Standard.

If you did not define special properties of [localeID] in the default.ini, registerosimage will add default properties:

```
[localeID]
 Type=Text
 MaxLength=255
 Trans=yes
 Comment=CAMENU\xxxx
```

Using this specification, the parameter presents a text field (Type=Text) of size 255 (MaxLength=255). It can also inherit values from predecessor configurations (Trans=yes). The comment specifies the location of the file where the parameter has first been detected.

Note: Do not omit the dollar characters ($) before and after the parameter name. They are mandatory.

Template.ini

The IPS comes with a set of OS image templates you can see during the usage of createosimage. All available images are defined in the Template.ini, which controls the createosimage command.

Creating new image types and new installation templates are for experienced users only.

New types can be added by adding an [ostype] section, including the keys for specification. See key descriptions below.

Template files in os-template\camenu\<ostype>.xx and in os-template\images\<ostype> also belong to an OS type. The assignment is defined in the template.ini file.

[<ostype>]

Section that specifies the template of the OS type.

typecomment = <string>

The string is only shown in the usage of createosimage as comment of the type.

imagetemplates = <ostype>

Name of the directory which contains the template files under \images.

identfile = <file>

Specifies a file that must be in available on the CD image in order to check whether it is the correct CD. This specification is used if the OS is on one CD, like most Windows operating systems, and morethanonecd is not set.

Identfile can also be used in the <copysection> to identify more CDs.

Default: Empty (no check will be done)

cdrootpath = <directory>

Default subdirectory on CD1 where the source root of the OS files is found.

Cdrootpath is also used to find the file <identfile> on CD1 when specified in the [<ostype>] section.

Cdrootpath has no effect if the -s <OSpath> option is given.

If a <copysection> is specified for CDs, the <copysection> entries are used to copy the directories from the CD or to find the <identfile>. Cdrootpath does not extend the specified copy paths or <identfile> from the <copysection>.

Default: ” .”

ossubpath = <directory>

Specifies a destination subpath of <imagename>\<imagename>\<subpath> for template files from os-template\images\<ostype>.

The OS files will be copied below that <subpath> directory, too.

If a <copysection> is specified for CDs, the <copysection> entries are used to copy the directories from CD. Ossubpath does not extend the specified copy paths from the <copysection>.

Default: No additional subdirectory

batfile = <file>

Name of the DOS installation script for installation using MS SHARE to create camenu\<imagename>.bat. <file> is the name of a template file in os-template\camenu.

tftpfile = <file>

Name of the DOS installation script for installation using TFTP to create camenu\<imagename>.ftp. <file> is the name of a template file in os-template\camenu.

createzip = <yes,no>

If this entry is set to yes, the -z option of createosimage also creates a <imagename>.caz file from the OS data. The files are stored in the managedpc\images\<imagename>\<imagename> directory.

parameterdefinition = <file>

Name of the file that contains the template of default parameter definitions used for default.ini. <file> is the name of a template file in os-template\camenu.

responsefile = <file>

Name of the unattended response file template to create camenu\<imagename>.inf. <file> is the name of a template file in os-template\camenu.

fileswithparameter = <file>

Name of the template list of files, including parameters that can be changed with the DSM Explorer. The template is used to create the osinfo.ini file. <file> is the name of a template file in os-template\camenu.

partitionfile = <file>

Name of the disk partition schema to create camenu\<imagename>.par. <file> is the name of a template file in os-template\camenu.

stringtosubstitute = <string>

String that will be replaced by <imagename> in camenu\<imagename>.bat and camenu\<imagename>.ftp.

imagefileextension = <string>

Name of an image file extension, for example, <[set the File Name variable]>.gho, to find the image file below the -s <OSpath>.

destdriverpath = <string>

Name of the folder on the target system where the drivers are stored.

Examples: oem\driver

sdostype= <internal number>

As it is possible to predefine a DSM PXE target, activating an OS installation and SD install jobs, the DSM manager needs to know what SD type the assigned OS is.

Example: 12

createshare = <string>

Specifies that a share should be created on the IPS and on a boot server. Valid values are as follows:

MS=MS-share

NFS=NFS-share

MSNFS=both

MSW=writable MS-share

Default: MS

finaljobstate = <0|1> (0=current;1=stopped)

Defines the final job state after boothd or registration of the DSM agent.

writabledir = <drirectory>

Defines the write-enabled image share for canonprv if createshare=MSW (getimagexxx).

NOopt... = <0|1> (0=visible,default; 1=skipped)

Skips or shows the corresponding IPS wizard page.For example, if you specify NOoptProdKey =1, the Product Key page in the New OS Image wizard will be skipped. You can use the following parameters to skip the wizard pages: NOoptOSPath, NOoptShare, NOoptProdKey, NOoptLocale, NOoptRespFile, NOoptDriver.

imgtoolcopyN = <imgtoolsubdirectory>:<file1>[><file1renamed>];[<file2>[><file2renamed>]]...

Defines the imaging tool files that you want to add to the image.

sysprepcopyN = <OStype>@<sysprepsubdirectory>[:<file1>;[<file2>]]...

Defines the sysprep files that you want to add to the image.

imageflags = <list of image flags>

Defines the image flags for checking the boot image compatibility with the OS image. Possible values are F140-2, DosTftp, DosLm, PeTftp, PeLm, Pe2Lm, Pe64Tftp, Pe64Lm, DosLmT

batfileWpe = <file>

Defines the name of the WinPE installation script for installations using a share to create camenu\<imagename>.cmd

tftpfileWpe = <file>

Defines the name of the WinPE installation script for installations using TFTP to create camenu\<imagename>.ftw

partitionfileWpe= <file>

Defines the name of the disk partition template to create from WinPE camenu\<imagename>.wp1

copyfrompath=<copysection>

Defines the name of a <copysection> that defines the files and directories to copy from a source path containing the Linux distribution files. See createosimage -i<name> -o<type> -s <OSpath>.

The following keys describe the number of CDs required to copy all files and directories into the OSIM OS-image:

morethanonecd = <number>

Defines the number of CD's to copy. Each CD has a key cd1,cd2,cd3.. that points to a special <copysection> that lists all files, directories to copy.

cd1,cd2...cdn = <copysection>

Specifies the name of the copy section.

The following entries are used with the -a <sharename> option. That means the install files must be copied from CD described in the CD100, … section must be copied in the OS image because it includes installation procedures, but the other OS files from the Linux CDs are on a central NFS Server on a NFS share (createosimage -i <name> -o <type> -a <sharename>).

copytemplatesalways = <yes|no>

Specifies whether the templates <imagetemplates> should be copied in the <imagename>\<imagename>.

If this value is set to yes, createosimages -i <name> -o <type> -a <sharename> copies the OS specific template files into the image even if the OS files itself located on another share.

Createosimages -i <name> -o <type> -a <sharename> can also copy files from the Linux CD to this minimum image. Usually, these are normally the Linux loader and other files needed in the DOS phase.

morethanonealwayscd = <number>

Defines the number of CD's to copy even if the -a switch is used. Morethanonealwayscd is only checked if copytemplatesalways = yes. All CDs are defined in one or more <copysection>. With morethanonealwayscd, createosimage looks for keys starting with cd100,cd101,cd102=<copysection>.

cd100,cd101...cd10n = <copysection>

Specifies the name of the copy sections.

[<copysection>] =

This section describes the directories or files to be copied from this CD.

<cddirectory>=<imagedirectory>

Copies cddirectory to imagedirectory (or <cdfile>=<imagefile> copies cdfile to imagefile). You can use wildcard characters such as * and ?.

The source path must start from CDROOT but without the drive name, for example, d:\

The destination path is relative to the shared directory of the image.

Append the source file to an existing destination file:

```
<cdfile> => <imagefile>
```

Read the source file from an existing image instead of reading from the CD:

```
< <sourcefile> = <destinationfile>
```

Read the complete CD, DVD into an iso file:

```
DVD:: = <file.iso>
```

Extract a specified package file from the rpm package on CD,DVD to the destination:

```
RPM::<source package file.rpm> = <destination file name>
```

The list of supported OS types is specified in the template.ini. For a complete list, see the Usage of CreateOSImage.

Note: If the IPS is also a Boot Server, CreateOSImage will create an OS image share. Depending on the “createshare” parameter CreateOSImage will also create an NFS share if the IPS system provides a NFS service.

Register Operating System Data of External Images

In addition to the CA OS images created with the createosimage command, external OS images can be registered to the domain manager. The manager however, does not check whether the image is on the target boot server.

Note: External OS image names are not restricted to eight characters (8.3 notation).

Note: Either an internally stored OS image or an external OS image must be registered in the management database (MDB) before an operating system installation job can be created for a target computer.

To provide an operating system description and register it in the management database (MDB)

1. Create a new directory with the OS image name, for example, winxppro, as follows:

```
mkdir DSM-installdisk:\Program
Files\CA\DSM\Server\SDBS\var\managedpc\images\winxppro
```

2. Copy default.ini and osinfo.ini from the DSM-installdisk:\Program Files\CA\DSM\osimips\os-template\images\os-image\ directory into the following directory:

   DSM-installdisk:\Program Files\CA\DSM\Server\SDBS\var\managedpc\images\<image name>

3. Open the default.ini file from the following location using an editor:

   DSM-installdisk:\Program Files\CA\DSM\Server\SDBS\var\managedpc\images\<image name>\default.ini

4. Change the OSImage assignment as follows:
   `OSImage=winxppro`

5. Open the osinfo.ini file at the same location using an editor:

   DSM-installdisk:\Program Files\CA\DSM\Server\SDBS\var\managedpc\images\<image name>\osinfo.ini

6. Change the imagename assignment as follows:

   `imagename=winxppro`

7. Register the operating system data in the MDB using the registerosimage command as follows:

   ```
   registerosimage -b -i winxppro -s domain_manager_name
   ```

   The operating system data of the external image, winxppro, is now registered to the management database of the domain manager.

Language Identifiers

The language identifier specifies the code page to use during the operating system setup. Valid values are as follows:

CHS

Chinese (PRC, Singapore)

CHT

Chinese (Taiwan, Hong Kong SAR, PRC)

ENU

Latin (North America, Western Europe). This is the default value.

HEB

Hebrew

JPN

Japanese

KOR

Korean

ROM

East European (Romania, Hungary, Poland, Slovakia, Czech Rep.)

THA

Thai

Define Disk Partitions for WinPE-based Windows Images

A WinPE-based OS image has its own disk partition file named <imagename>.wp1 in Microsoft diskpart format. The disk partition file is created in the camenu directory of the image store. You can specify which disk partition to use, for installing the OS image on the target computer.

To define the disk partition for WinPE-based OS installations

1. Deploy the OS image on the target computer in DSM Explorer.

   An OS Installation job is created under All Computers, {Group Details}, OS Installations.

2. Navigate to All Computers, {Group Details}, OS Installations, and select the OS installation job to which you want to specify the disk partition.

   OS installation parameters appear in the right pane.

3. Double-click the WinPEPartition parameter.

   The Edit OS Installation Parameter dialog opens.

4. Select one the following values:

   Use default partition file

   Creates only one active primary c: partition on disk 0 of the target computer.

   Use special partition file <imagename>.wp1 from CAMENU

   Creates the hard disk partitions as defined in the .wp1 file.

   When OS installation job executes at the target computer, the operating system is installed on the disk partition you specified in the WinPEParitition parameter.

   Note: If the boot parameter "CreateNewPartitions" is set to true , OSIM ignores the SANID boot parameter because it uses the existing available partition (C drive) created by previous windows installation for new OS Installation. Supplying boot parameter SANID with boot parameter "WinPEPartition" with value "Use special partition file W2012.wp1 from CAMENU", forces OSIM to ignore parameter "WinPEPartition" value. If SANID is specified, OSIM identifies the disk needed to be used for OS installation and ignores the supplied external partition file for OS deployment.

Chapter 5: Using GHOST (DOS) Imaging Tools

This chapter informs you how to use images created with the GHOST imaging tool.

You will learn how to create images that use GHOST disk images.

This section contains the following topics:

Create OS Images That Use GHOST (DOS) Disk Images

Note: The following example describes how to create OS images for Windows 2000 that use GHOST disk images. The creation of Windows XP and Windows 2003 images is working analogously.

1. Install the master Windows XP, or Windows 2003 on a computer and add all service packs needed for the installation.

   The system drive for the installation should be drive C: and the format of the type should be FAT16/32. If you use the 32-bit version of GHOST, type NTFS is also possible.

   If a FAT16 system partition is used to create the GHOST image, its size must not exceed 2 GB.

   Use FAT32 if the OS prototype installation is larger than 2 GB and DOS GHOST is used.

   If the GHOST image is created from the system partition only, existing partitions at the target machine other than the system partition are preserved.

   In the case of DOS GHOST, the FAT16/32 partition will later be automatically converted to NTFS and expanded to the end of unused disk space. That means, the FAT 16/32 system partition from the GHOST image will normally be expanded to the size of the previous system partition.

   If (CreateNewPartitions=yes), the whole disk will be used.

2. We recommend that you not install any additional applications. Because the OSIM will also install a DSM Software Delivery agent, this should be left to Software Delivery.

3. After the installation of the master Windows is complete, apply sysprep to the model system. Create the directory C:\sysprep and copy the files sysprep.exe and setupcl.exe into this directory.

   The files can be extracted from the cab file SUPPORT\TOOLS\DEPLOY.CAB on the OS installation CD.

   There are small differences between WinXP, 2003. Please read the sysprep documentation.

   Note: Service packs could release new versions of sysprep, which must be used after the server pack was applied.

4. Run sysprep on the model computer.

   - For Windows XP, or Windows 2003:
     Open Start, Run, execute

     ```
     c:\sysprep\sysprep.exe -mini -reseal -quiet
     ```

   - For Windows Vista:
     Open Start, Run and execute.
     Then:

     ```
     cd c:\windows\system32\sysprep
     ```

     and call

     ```
     sysprep.exe /oobe /generalize /quiet /shutdown
     ```

   Important! After system shutdown, do not restart the computer from hard disk!

5. If you use DOS GHOST, insert a DOS boot floppy into the model system and start the system from the floppy.

6. After starting DOS, insert a floppy with the GHOST program and create the ghost image.

7. Provide a GHOST image of the system, for example, Win2k.gho, and store it in a location accessible to the Image Prepare System.

   Assume that this is \\ips\win2k, which is on the Image Prepare System at c:\win2k.

   Note: If you use DOS GHOST, the name of the GHOST image must be a valid 8.3 DOS file name.

8. Copy the ghost DOS client “ghost.exe” to the c:\win2k directory, also.

Now you can build the GHOST-based OSIM image.

Build the GHOST (DOS)-based OS Image

To build a GHOST-based OS image, launch the command prompt using the Start button and enter the following command at the DOS prompt:

```
createOSImage -i Win2K -o GHOST-W2K -s c:\win2k -k productkey
```

When the command successfully completes, the image is stored at the MANAGEDPC\IMAGES directory relative to the installation directory of CA\DSM:

- The files Win2k.par, Win2k.inf, and Win2k.bat are stored at MANAGEDPC\CAMENU.
- The OS image provides a default installation of Windows 2000, Windows XP, or Windows 2003 according to the prototype installation, and includes a default set of parameters that can be set from the OS Installation Manager.
- The OS image provided will work with the default boot images, osinstal.2 and osinstal.3; no special boot images are needed.

Register the OS image on the OS Installation Manager

After customization is completed, the OS Image can be registered at the OS Installation Manager and its associated domain manager, named myLocalServer.

You can use the OS Image Registration wizard of the DSM Explorer
(DSM Explorer / <domain manager> / Software / Boot and OS Image Library / Image Prepare System / Wizards)
or the registerOSImage command directly from the command prompt. Use the following command:

```
registerOSImage -s myLocalServer -i Win2k
```

When the command successfully completes, the image is registered with the name Win2k at the OS Installation Manager.

It is ready for operation. At the associated domain manager, the image is registered as Win32k with a version number of 11.1.0/00.

It is ready for deployment like any other OS image.

Note: You have to register an OS image before you can address it at a domain manager or OS Installation Manager. When using FAT32, the boot images osinstal.2 and osinstal.3 must be built from a Win98 DOS system.

Using the Command Line Interface

cadsmcmd is the command line interface that lets you automate processes that drive Client Automation. The following list shows the OSIM-related cadsmcmd commands and the corresponding tasks :

- targetcomputer—Managing Target Computers
  - create—Create Target Computer
  - delete—Delete Target Computer
  - list—List Targetcomputer
  - modify—Modify Target Computers
  - showAttr—Show Attributes of Target Computer
  - activateOS—Activate OS Installation Order
  - cancelOS—Cancel OS Installation Order
  - deletePlannedOS—Delete Planned OS Installation Order
  - deleteScheduledOS—Delete Scheduled OS Installation Order
  - modifyInstallParameter—Modify Install Parameters of OS Installation Orders
  - modifyOS—Assign OS Image to Target
  - reactivateOS—Reactivate Stopped / Failed OS Installation Order
  - reinstallOS—Re-install Operating System
  - removeFromOsim—Remove Targets from OSIM
  - showInstallParameter—Show Install Parameters of OS Installation Orders
- bootserver—Managing OSIM Boot Servers
  - assignTargets—Assign Targets to Boot Server
  - list—List Boot Servers
  - listImages—List Images at Boot Server
  - listTargets—List Targets Registered at Boot Server

- image—Managing OS Images
  - assignToTargets—Assign OS Images to Targets
  - delete—Delete Images from the OSIM Database
  - list—List Images Registered at OSIM Manager
  - listConfigs—List Targets and Configurations of OS Images
  - listDeployed—List Boot Servers where OSIM Image Is Deployed / Removed
  - modify—Modify OSIM Image
  - deploy—Deploy OSIM Images to Boot Servers
  - remove—Remove OSIM Images from Boot Servers
- managedcomputer—Manage Computer Configurations
  - listConfigs—List Computer Configurations
  - reportCurrentConfig—Report Current Computer Configuration
  - requestConfigReport—Request a Configuration Report

Example: CADSMCMD Commands for Making Unmanaged Targets Managed

You can manage your OS installation either through DSM Explorer or through a CLI. The following cadsmcmd command allows unmanaged but named target computers to become managed by the OS Installation Manager:

```
cadsmcmd targetComputer action=setupOS
```

Assume a new PXE-enabled system installed with an agent has been registered at the related OS Installation Manager and the associated domain manager as D-PB-0103, but the system is still unmanaged by the OS Installation Manager. Enter the following command to assign the OS image WinXP as a planned OS installation to the target computer so that it gets managed by the OS Installation Manager:

```
cadsmcmd targetComputer action=setupOS name=D-PB0103 OSImage=WinXP
```

Note: For more information about the usage of this command, see the *CLI Reference Guide*.

Chapter 6: OSIM Images Using ImageX and Ghost32

OSIM supports the following images created using ImageX and Ghost32:

- For the list of supported operating systems for ImageX-based images, see the Compatibility Matrix.
- ImageX-based images
- Ghost32-based images

  Note: If you are using GHOST (DOS) imaging tools, see the chapter "Using GHOST (DOS) Imaging Tools".

You can create ImageX images in two different ways:

1. By using an OSIM support image and an OSIM WinPE 2.0 image (see page 113)
2. In a manual way using a WinPE 2.0 boot CD (see page 127)

Requirements

- Image Prepare System with Boot Server on the same system
- ImageX requires MS Windows AIK installed on the Image Prepare System
- Ghost32 support requires the 32-bit-executable file from Ghost 8 or Ghost 11 (Ghost Solution Suite 2.0). No Ghost server components have to be installed.

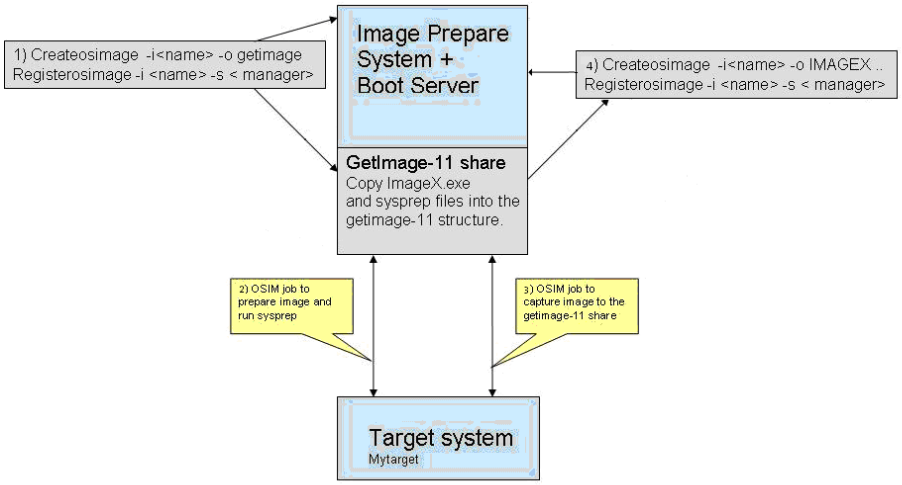
Steps to Create an ImageX OSIM OS Image Using an OSIM Support OS Image

This section describes the steps to create an ImageX or Ghost32 OSIM OS image using an OSIM support OS image.

Firstly, a functional overview (see page 114) is given, then the major steps are presented:

- Step 1: Create a Support-OS-Image (see page 115)
- Step 2: Add Sysprep Files to the OS Image Structure
- Step 3: Prepare and Capture ImageX or Ghost Images (see page 121)
- Step 4: Create and Register the Captured Image as OSIM OS Image (see page 124)

Furthermore there is information how to add drivers to the Windows based OSIM OS Image (see page 125) and a description of the new alternative partition definition file (see page 133). The following diagram depicts the steps to create an OSIM OS image using an OSIM support OS image:



Support OS Functional Overview

A support operating system makes it easier to prepare and capture ImageX or Ghost images from a PXE-enabled model computer by executing an OSIM job and using an OSIM Windows PE20 or Windows PE30 boot image.

See Compatibility matrix for latest operating system support.

You can create and use the above support OS types only if Image Prepare System, the Boot Server, and DSM Explorer are on the same computer. The Boot Server must be configured for Share Access using the **sdbsswitch** command.

- Use the command **sdbsswitch –l,** you will find out the current configuration.
- Use the command **sdbsswitch –s** to switch the boot server to share mode.

For the list of GETIMAGE supported operating system, see the Compatibility Matrix.

The GETIMAGE-support-operating system provides an OS-share with directories for the imaging tools like Ghost and ImageX, and it provides directories for the different OS-specific sysprep-programs.

The share also has a writable subdirectory for the images to capture.

Create a Support-OS-Image (Step 1)

To create a support image of the GETIMAGE type use the IPS commands with the following parameters:

```
createosimage –i <imagename> -o <support_OS_type> [-g <imgtools>]
```

If you want to use only ImageX to capture a model computer and Microsoft Windows AIK is installed on the IPS, you do not have to provide the -g option. The ImageX files will be copied automatically into the GETIMAGE-structure. If you want to use Ghost32 as the imaging tool, you have to provide the -g option with the path to ghost32.exe client program.

To register the support image, use the following command:

```
registerosimage -b –i <imagename> OS -s <manager>
```

ImageX and Ghost32 in the GETIMAGE-Structure

If do not have Windows AIK installed or you have not provided the path to the imaging tool during the creation of support OS image, you can copy the required files to the GETIMAGE-structure manually.

ImageX

If you are using ImageX, copy the ImageX-files

- imagex.exe
- intlcfg.exe
- wimfltr.inf
- wimfltr.sys
- wimgapi.dll

from the Windows AIK installation directory

```
C:\Program Files\Windows AIK\Tools\x86\
```

to the OSIM image store:

```
<OSIM-image store>\<imagename>\<imagename>\imagex\
```

Ghost32

If you are using Ghost32, copy the 32bit-ghost.exe to

```
<OSIM-image store>\<imagename>\<imagename>\ghost32\ghost.exe
```

GETIMAGE64

This release provides the following new GETIMAGE type to support 64-bit version of WinPE 4.0.

GETIMAGEX64

Captures Windows 7x64, 8x64, 8.1x64, Windows Server 2008x64, 2008 R2, 2012, 2012 R2 OS images.

ImageX

This release provides the following new ImageX types to support 64-bit version of WinPE 4.0.

IMAGEX64-WIN7

Supports Windows 7x64, and Windows 7 SP1x64 OS images.

IMAGEX64-WIN8

Supports Windows 8x64 and Windows 8.1x64 OS images.

IMAGEX64-W2008/R2

Supports Windows Server 2008x64 and 2008 R2.

IMAGEX64- W2012/R2

Supports Windows Server 2012, 2012R2.

(Optional) Backup your model PC (Step 2)

The process of prepare (sysprep Step3) modifies the OS installation on the model computer such that it cannot reboot into the operating system anymore. You can back up the model computer so that you can restore the same and boot into the operating system after you have captured the image.

To backup an image, you must run an OSIM backup job that captures the OS image using the specified imaging tool.

To create OSIM install jobs to back up an image:

1. Create an OS installation job for the model computer in DSM Explorer.
2. Set the value of following OS installation parameters:

Parameter	Description
OSImage	Specify "getimg" or "getwepos" depending on the support OS type.
BootSpecial	Specify the required boot image.

Parameter	Description
ImagexImage	Specify the name of the image file to create.
CreateImage	Specify "backupImageX" or "backupGhost"
BackupPolicy	Specify the backup policy (see page 120) for the job.

3. Activate the job (use Wake-On-LAN for automatic boot).

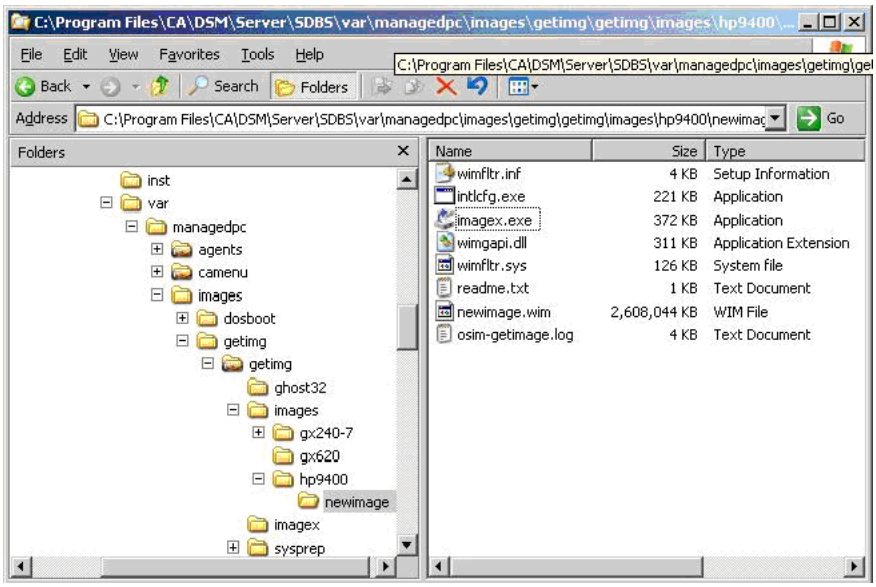
   The following events occur:

   - The model computer boots WinPE boot image again and executes the getimage.cmd script from the camenu share.
   - The getimage.cmd command shares the getimage-11 or getimage-wepos-11 share, depending on the OS type you selected. The images subdirectory is prepared to hold all images. The getimage.cmd command looks for a subdirectory with the name of the model computer under the images folder. If the subdirectory does not exists, getimage.cmd creates the directory. All images from a model computer are stored in the subdirectory with the model computer name.
   - Further on, getimage.cmd looks for a subdirectory with the name of the image ($ImageXImage$) under the model computer subdirectory. If this directory exists, it extents the name to $ImageXImage$n (n=1-5) and tries to create a new subdirectory again. If the directory creation is successful, getimage.cmd stores the new image in the subdirectory.

     Note: The success of directory creation depends on the BackupPolicy that you have set. If the BackupPolicy is set to write images and wrap around, getimage.cmd overwrites the oldest image when the defined number of images already exists. If the BackupPolicy is set to write images and not wrap around, getimage.cmd aborts with an error message that the directory name already exists.
   - getimage.cmd copies the ImageX client to the new image directory, executes ImageX to capture c: partition of the model computer, and stores the images as a .wim file in the following path:

     ```
     \\<OSIM-image store>\<imagename>\<imagename>\images\<computername>\
     <$ImageXImage$n>\<$ImageXImage$n>.wim
     ```

   The following screenshot shows the location of the new image:



Note: If the C: partition does not exist on the model computer, the getimage.cmd command terminates after writing to the error to the log file.

For Ghost32, the directory contains the ghost.exe file and the file <$imageXImage$>.gho will be created by getimage.cmd.

4. Finally the model computer shuts down and the OSIM job status in DSM Explorer is set to Stop.

Backup Policy

Backup policy defines the backup strategy for the BackupImageX and BackupGhost jobs for one target computer. You can define the backup policy using the $BackupPolicy$ parameter. Backup jobs create a new image directory with the name of the image as specified in the $ImageXImage$ parameter. You can configure the backup policy to specify the number of backup images that you want to store for one target computer, and the action required when the number of backup images exceed the configured value.

Following are the valid values for the $backuppolicy$ parameter:

write up to 5 images, do not wrap around [0]

Specifies that every backup job using the same image name creates a new image directory by appending 1 to 5 to the directory name. After creating $ImageXImage$5, the backup job fails.

write 1 image and overwrite [1]

Specifies that every backup job using the same image name overwrites the existing image.

write 2 images and wrap around [2]

Specifies that backup jobs using the same image name create up to two images and then overwrite the oldest.

write 3 images and wrap around [3]

Specifies that backup jobs using the same image name create up to three images and then overwrite the oldest.

write 4 images and wrap around [4]

Specifies that backup jobs using the same image name create up to four images and then overwrite the oldest.

write 5 images and wrap around [5]

Specifies that backup jobs using the same image name create up to five images and then overwrite the oldest.

Note: Choosing a wrap around action overwrites the oldest of the images when the number of backups exceed configured value. The first backup image directory gets the name from the $ImageXImage$ parameter and the succeeding image directories get the name from $ImageXImage$ with a trailing number (1-4).

More information:

Create OSIM Install Jobs to Capture an Image (Step 3b) (see page 122)

Logfiles of the Getimage support OS job.

Getimage.cmd creates for each getimage job one logfile on the IPS/boot server in

```
\\<OSIM-image store>\<imagename>\<imagename>\images\<computername>\
<$ImageXImage$n>\osim-getimage.log
```

The logfile contains information about the prepare phase executed from runonce (osimsysprep.cmd) and information about the backupImageX, backupGhost phase executed from getimage.cmd.

Prepare and Capture ImageX or Ghost Images (Step3)

The following steps must be done for every new ImageX or Ghost image.

Install the Model Computer

Your model computer can be installed from CD, DVD, or using OSIM.

Note: Do not install any CA agents, because this is part of the OSIM ImageX installation.

If you install the model computer using OSIM, set the OSIM job parameter

```
installAgent=no
```

```
BitLocker=no
```

Create OSIM Install Jobs to Prepare an Image (Step 3a)

To create OSIM install jobs to prepare images:

1. Create a getimage OS installation job for the model computer in DSM Explorer.
2. Set the value of following OS installation parameters:

Parameter	Description
OSImage	Specify "getimage" or "getwepos" depending on the support OS type.
BootSpecial	Specify the required boot image.
SysprepVersion	Specify the name of the directory that contains sysprep files
CreateImage	Specify "prepare"

3. Activate the OSIM job and boot the model computer.
   The following will happen:

   The model computer boots into WinPE boot image and the job execution script <image>.cmd copies the sysprep files from \\<OSIM-image store>\<imagename>\<imagename>\sysprep\$SysprepVersion$ to the hard disk of the model computer. It also copies the OSIM prepare script “osimsysprep.cmd” and the driver registry file and adds execution of the script to the registry (runonce) of the Model computer OS.

   Finally the script reboots the target into the OS of the Model computer.

   Note: No DSM agent should be installed on the Model PC. An installed DSM agent can’t be removed from the getimage (prepare) function.

4. When the model computer is up again, logon as administrator.
   The following will happen:

   Runonce launches osimsysprep.cmd. Osimsysprep.cmd removes OSIM files from previous OS installations, adds the OSIM driver path to the registry and executes sysprep.

   Sysprep shuts down the model computer.

   On the DSM Manager the prepare job becomes stopped.

Create OSIM Install Jobs to Capture an Image (Step 3b)

You must run another OSIM backup job to capture the OS image using the specified imaging tool.

To create OSIM install jobs to capture an image

1. Set the value of following OS installation parameters:

Parameter	Description
OSImage	Specify "getimage" or "getwepos" depending on the support OS type.
BootSpecial	Specify the required boot image.
ImagexImage	Specify the name of the image file to create
CreateImage	Specify "backupImageX" or "backupGhost"
BackupPolicy	Specify the backup policy (see page 120) for the job.

2. Activate the job again (use Wake-On-LAN for automatic boot).

   getimage.cmd copies the ImageX client to the new image directory, executes ImageX to capture c: partition of the model computer, and stores the images as a .wim file in the following path:

```
\\<OSIM-image store>\<imagename>\<imagename>\images\<computername>\
<$ImageXImage$n>\<$ImageXImage$n>.wim
```

After the image is captured successfully, OSIM job status in DSM Explorer is set to Stop. You can now create and register the OSIM OS master image for OS deployment. If you have taken a backup of the model computer, you can restore it now.

More information:

Backup Policy (see page 120)

(Optional) Restore the OS Image on the Model Computer (Step 4)

To restore the OS image on the model computer

1. Set the value of following OS installation job parameters:

Parameter	Description
OSImage	Specify "getimage" or "getwepos" depending on the support OS type.
BootSpecial	Specify the required boot image.
ImagexImage	Specify the name of the backup image file to restore
CreateImage	Specify "restoreImageX" or "restoreGhost"
RestorePolicy	Specify the restore policy (see page 124) for the job.

2. Activate the job (use Wake-On-LAN for automatic boot).

   The job restores the OS image on the active partition of the model computer. After the image is restored, the model computer reboots and the status of the job is set to Stop in DSM Explorer. You can view the log file at <OSIM-imagestore>\images\<imagename>\<imagename>\images\<computername>\<$ImageXImage$n>\osim- restore.log

   Note: If there is no active partition on the model computer, the job creates an active partition using the <OSIM-image store>\images\<imagename>\<imagename>\images\<name_of_model_PC>\$ImageXImage>\$ImageXImage$.wp1 diskpart file, and installs the image in this new partition.

More information:

Restore Policy (see page 124)

Restore Policy

Restore policy defines the restore strategy for the RestoreImageX and RestoreGhost jobs for one target computer. You can configure the restore policy to specify the backup image that you want to restore on the target computer.

Following are the valid values for the $restorepolicy$ parameter:

Take the backupimage as named in $ImageXImage$" [0]

Specifies that the backup image with the name equivalent to the value of $ImageXImage$ parameter be restored on the target computer.

Take the newest backupimage with the base name $ImageXImage$" [1]

Specifies that the latest backup image with the base name equivalent to the value of $ImageXImage$ parameter be restored on the target computer. The backup image can have a trailing number if you have more than one backup image.

Take the second newest backupimage with the base name $ImageXImage$" [2]

Specifies that the next-to-last backup image with the base name equivalent to the value of $ImageXImage$ parameter be restored on the target computer. The backup image can have a trailing number if you have more than one backup image.

More information:

(Optional) Restore the OS Image on the Model Computer (Step 4) (see page 123)

Create and Register the Golden Image as OSIM OS Image (Step 5)

After a successful prepare and capture step you can create and register the golden image for multiple deployment.

To create and register the captured image as an OSIM OS image, use the following two commands:

1. Use the IPS command createOSImage to create an OSIM OS image:

```
createOSImage -i <OSIM image name> -o IMAGEX-XP (or GHOST-XP)
 -s \\<OSIM-image
store>\<imagename>\<imagename>\images\<computername>\<$ImageXImage$n>
[-k <product-key>]
```

2. Then register the image to the domain manager:

```
registerOSImage -i <OSIM image name> -s <name of the manager> [-b]
```

Now the new OS image can be used for target installation with OSIM.

Add drivers to the Windows based OSIM OS Image

The OSIM- ImageX and -Ghost images of Windows OS provide a directory “driver” for additional PNP driver which needs to be recognized in the (mini) setup of the operating system on the target.

The driver directories are prepared in the new ImageX and Ghost images in the IPS image store.

Required drivers can be inserted on the IPS either by copying the files into the driver directory of the OSIM OS image or by using the command:

```
createosimage  ……. –t <path to driver> …..
```

Drivers also can be added to an OSIM OS SD package in the Software Package Library using the DSM Explorer.

If a boot server is configured for share mode, the driver can be copied in the related directories in the boot server image store.

The following driver directories will be prepared in the OSIM images, where <path to image> stands for the path listed by createosimage –x .

```
<path to image>\<type IMAGEX-OSName>\oeminst\driver[\subdirs]
```

The additional drivers will be recognized in the hardware-analyzing phase. Therefore they must be plug-and-play drivers.

Additional Windows Vista, Windows 2008, and higher version drivers will be inserted in the WindowsPE-mode of the installation.

The Windows Vista, Windows 2008, and higher version drivers must also be plug-and-play drivers and they must be certified for these Windows OS. You can store these Windows OS drivers in a directory under \oeminst\driver.

Add Additional Drivers to Windows Embedded POS Images

You must add additional drivers to Windows Embedded POS images manually, because IPS does not add them to the image automatically at the time of creating the image.

To add additional drivers to Windows Embedded POS images

1. Copy the driver files (.inf, .sys) of the additional drivers to the <OSIM-image store>\images\<imagename>\<imagename>\D2DriverFiles directory.
2. Open the driver description file driverinstall.xml under <OSIM-image store>\images\<imagename>\<imagename>\setup and include the description of additional drivers. The driver description is available in the .inf file of the driver.

   Note: The detailed definition of driverinstall.xml schema is available in the Windows Embedded POS documentation.

   The following snippet shows a sample driver description:

```
<?xml version="1.0" encoding="utf-8" ?>
<Drivers>
  ……..
    <AdditionalDevice ClassId="{4d36e972-e325-11ce-bfc1-08002be10318}"
SourceDir="\D2DriverFiles">
    <!-- "Broadcom, NTx86.5.1, NTamd64" -->
    <File Name="b57win32.inf" InstallDir="inf" />
    <File Name="b57xp32.sys" InstallDir="system32\drivers" />

    <HardwareId Id="PCI\VEN_14e4&amp;DEV_1677&amp;SUBSYS_01AD1028&amp;REV_01"
/>
    </AdditionalDevice>
</Drivers>
```

3. Save and close the file.

   The additional drivers are added to the image. These drivers will be installed on the target computers when the image is applied.

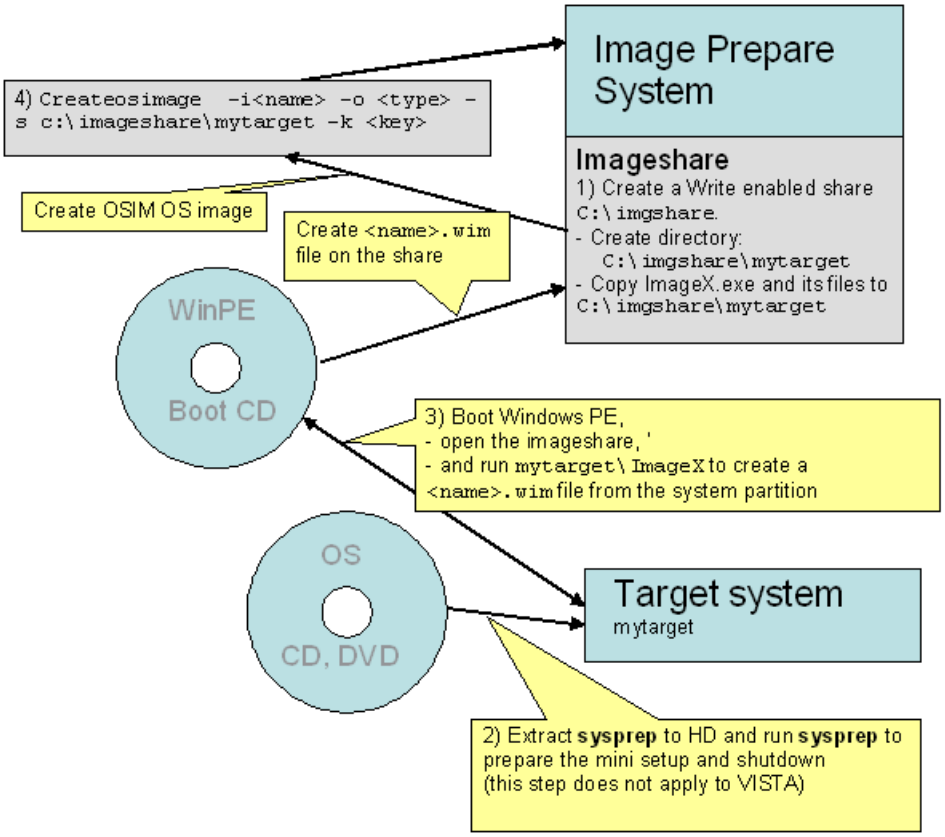
Steps to Create an ImageX OSIM OS Image Manually Using a WinPE Boot CD

The ImageX support works similar to the Ghost32-support. The use of the Sysprep command and the capturing of images require manual steps.

This section contains the following topics:

Installation of the Model Computer

The following diagram shows the steps involved in creating an ImageX OSIM OS image manually using a WinePE Boot CD:



Install the Model Computer from the OS setup medium. Do not install any CA Agents because this is part of the OSIM ImageX installation.

Create an Image Prepare System (IPS) Share to Write the Image Files to the IPS

To create an Image Prepare System (IPS) Share to Write the Image Files to the IPS:

1. Create Write-enabled share, for example, “imgshare” on the IPS.

   This share can be used for imaging of all targets and image types (ImageX, Ghost).

2. Create for each template target a directory with the name of the target, for example,

   ```
   mkdir \\imgshare\mytarget
   ```

   If you plan to capture and hold more than one image from one target, create subdirectories for every single image.

3. Copy the ImageX or Ghost.exe (32bit version) and its files into the directory that will hold the image file

   ```
   (<name>.wim or <name>.gho)
   ```

Create a Bootable Windows PE RAM Disk on CD ROM

Read the Microsoft documentation for more details, for example, "Booting Windows PE"

Create the Images

To create the Windows Vista, Windows 2008, Windows 7, Windows 2008r2, Windows 8, Windows 2012 ImageX images, Windows 8.1, Windows 2012 r2 ImageX images, proceed as follows:

1. Install the operating system (Windows 2008r2, Windows 8, Windows 2012, Windows XP, Windows Vista, Windows 2008, Windows 7) for the model computer and add all service packs needed for the installation.

   The system drive for the installation should be drive C: and the file system type should be NTFS.

   It is recommended not to install any further applications. Because OSIM will also install a DSM Software Delivery agent, this should be left to Software Delivery.

2. When the installation of the model computer is complete, apply sysprep there:
   - Create the directory C:\sysprep
   - Copy the files sysprep.exe and setupcl.exe into this directory.

   The files can be extracted from the cab file

   `SUPPORT\TOOLS\DEPLOY.CAB`

   on the OS installation CD.

   (This step is not necessary for Windows Vista and Windows 2008: Just call:

   `c:\windows\system32\sysprep\sysprep.exe`)

   There are small differences between Windows Vista, Windows 2008, Windows 7, Windows 2008r2, Windows 8, Windows 2012.

   Please consult the sysprep-documentation .

   Note: Service packs could include new versions of sysprep, which must be used when the service pack has been applied.

   (This step is not necessary for Windows Vista because it includes already:

   `c:\windows\system32\sysprep\sysprep.exe`)

3. Run sysprep on the model computer.
   - For Windows Vista and Windows 7, Windows 2008r2, Windows 8, Windows 2012:
     Open Start, Run and execute.
     Then:

     ```
     cd c:\windows\system32\sysprep
     ```

     and call

     ```
     sysprep.exe /oobe /generalize /quiet /shutdown
     ```

   Important! After system shutdown, **do not restart** the computer from hard disk!

4. Insert the bootable Windows PE CD and boot the target from CD.

5. Mount the IPS share, for example:

   “net use v: \\IPS-system\mytarget

   and change to the target subdirectory.

   Execute the ImageX or Ghost client.

6. Provide an ImageX or GHOST image of the system partition, for example,

   winxpsp2.wim

   in the target subdirectory on the share.

   Note: GHOST cannot be used for Windows 2008 and VISTA.

Note: If you use the ImageX tools to provision Microsoft Windows Server 2008 R2 and additional language packages are required to be added to the installation, then the boot image used must be derived from the version 2 AIK SDK (WinPE 3.0). If a WinPE 2.0 image is used, the language pack install fails because there is no pkgmgr.exe available on WinPE 2.0 boot images.

Note: Enabling BitLocker drive encryption during OSIM provisioning is not supported on Windows Server 2008 editions. The BitLocker option appears (with no value) during OS image creation due to definition files that are common with Windows 7 imaging.

Build the ImageX- or Ghost32-based OSIM OS Image

To build an ImageX- or Ghost32-based OSIM OS image, proceed as follows:

- On the IPS launch the command prompt using the Start button and enter the following command:

  Example for Win XP SP2:

  createOSImage -i winxpsp2 -o IMAGEX-XP (or GHOST-XP1)-s C:\imgshare\mytarget -k productkey

  When the command completes successfully, the image is stored in the MANAGEDPC\IMAGES-directory, relative to the installation directory of CA\DSM.

  The files

  - winxpsp2.wp1
  - winxpsp2.inf
  - winxpsp2.cmd
  - winxpsp2.wp1

  (see “alternative partition definition file”) are stored at MANAGEDPC\CAMENU.

- The OS Image Prepare System provides a default installation of ImageX-based Windows XP, Windows 2003, Windows Vista, Windows 2008, Windows 7, Windows 2008r2, Windows 8, Windows 2012, and Ghost32-based Windows XP, and Windows 2003, according to the prototype installation, and includes a default set of parameters which can be modified at the OSIM manager.
- The OS image provided will work with the default Windows PE 20 boot image of this DSM version.

Create and Register the Captured Image as OSIM OS Image

To build an ImageX or ghost32-based OSIM OS image, proceed as follows:

1. On the IPS launch the command prompt using the Start button and enter the following command:

   Example for Win XP SP2 named winxpsp2:

   ```
   createOSImage -i winxpsp2 -o IMAGEX-XP (or GHOST-XP) -s C:\imgshare\mytarget
   [-k productkey]
   ```

   When the command completes successfully, the image is stored in the MANAGEDPC\IMAGES-directory, relative to the installation directory of CA\DSM.

   The files

   - winxpsp2.wp1
   - winxpsp2.inf
   - winxpsp2.cmd

   (see "alternative partition definition file (see page 133)") are stored at MANAGEDPC\CAMENU.

2. The OS image Prepare System provides a default installation of ImageX-based Windows XP, Windows 2003, Windows Vista, Windows 2008, Windows 7, Windows 2008r2, Windows 8, Windows 2012 and Ghost32 –based Windows XP, Windows 2003 according to the prototype installation, and includes a default set of parameters which can be modified at the OSIM manager.
3. The OS image provided will work with the default Windows PE 20 boot image of this new DSM version.
4. Then register the Image to the domain manager

   ```
   registerOSImage -i winxpsp2 -s <name of the manager> [-b]
   ```

   Now the new OS- Image can be used for target installation with OSIM

New Alternative Partition Definition File

The default partition schema included in the WinPE boot images may not fit to the schema of the ImageX-.wim file. Therefore an alternative partition description can be used.

A new partition control file <imagename>.wp1 will be created with new ImageX images. The file is located in the camenu directory\share and contains Microsoft diskpart instructions.

New WinPE boot images look for a <imagename>.wp1 file first. If this file is available in camenu, the partition schema will be used instead of the default schema included in the WinPE image.

You can modify the <imagename>.wp1 according to the Microsoft diskpart documentation.

Changing the Auto Answer Files with Windows System Image Manager

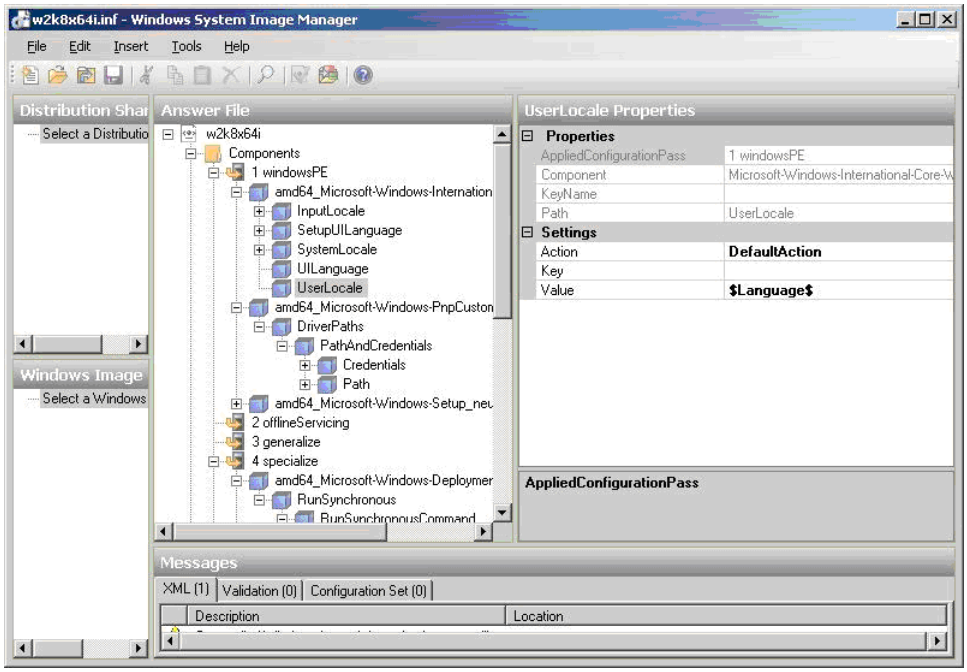
The auto answer files of the Windows VISTA, Windows 2008 or Higher Versions OS types can be modified with the Microsoft Windows System Image Manager.

After creating the OSIM OS image with “createOSImage” and before it is registered with “registerOSImage”, the auto answer file can be modified.

The auto answer files of the created OSIM OS images are stored in the OS image store in

```
C:\program files\CA\DSM\server\SDBS\var\managedpc\camenu\<imagename>.inf
```

The picture below shows the OSIM auto answer file of Win2008x64 with included OSIM OS install parameters.



It is possible to extend the OSIM auto answer file in accordance with the Microsoft specification.

For more information on how to add additional OS installation parameters $parameter$, see Add OSIM Parameters to Created Images (see page 26).

Add drivers to the Windows based OSIM OS Image

The OSIM- ImageX and -Ghost images of Windows OS provide a directory “driver” for additional PNP driver which needs to be recognized in the (mini) setup of the operating system on the target.

The driver directories are prepared in the new ImageX and Ghost images in the IPS image store.

Required drivers can be inserted on the IPS either by copying the files into the driver directory of the OSIM OS image or by using the command:

```
createosimage  …….. –t <path to driver> …..
```

Drivers also can be added to an OSIM OS SD package in the Software Package Library using the DSM Explorer.

If a boot server is configured for share mode, the driver can be copied in the related directories in the boot server image store.

The following driver directories will be prepared in the OSIM images, where <path to image> stands for the path listed by createosimage –x .

```
<path to image>\<type IMAGEX-OSName>\oeminst\driver[\subdirs]
```

The additional drivers will be recognized in the hardware-analyzing phase. Therefore they must be plug-and-play drivers.

Additional Windows Vista, Windows 2008, and higher version drivers will be inserted in the WindowsPE-mode of the installation.

The Windows Vista, Windows 2008, and higher version drivers must also be plug-and-play drivers and they must be certified for these Windows OS. You can store these Windows OS drivers in a directory under \oeminst\driver.

Chapter 7: WinPE Boot Images Support

This section contains the following topics:

Boot Image Considerations

- The DSM Explorer and the command line interface, cadsmcmd, provide integrated functions to handle Client Automation computers as OSIM PXE targets and launch OS installations to these targets.
- The boot image name parameters can be set at an operating system installation request.
- Boot images must be registered to the domain manager before they can be assigned from the DSM Explorer or command line interface.
- You can choose one of the registered boot images for each operating system installation request.
- The IPS provides Win32 tools for finding the boot server, loading files and images from the boot server, and tools to handle boot parameters within the Windows PE based OS installation.

Note: WinPE boot images must be prepared on the Image Prepare System. If not installed, upgrade the installation with Packaging Tools/OS Image Prepare System. This installs the Image Prepare System in \Program Files\CA\DSM\osimips.

The following sections describe how to create WinPE, WinPEx64, WinPE20, WinPE20x64, WinPE30, WinPE30x64, WinPE40, and WinPE40x64 boot images with Image Prepare System (IPS) commands or manually.

Creating Windows PE Images

This section describes how to create and use Windows PE images using one of the following methods:

- Image Prepare System (IPS) commands
- Manual steps

Creating Windows PE Images with the Command CreateBtImages

This section describes how to create Windows PE images with the IPS command createBtImages. We recommend using the latest WinPE boot images.

Creating Windows PE 2.0, 3.0, and 4.0 (32-bit) with CreateBtImages

To create a Windows PE 2.0, 3.0, and 4.0 32-bit ready to use with RegisterBtImages -i *name* - s *domain_manager*:

1. Install Windows AIK, or Windows ADK on the IPS box based on the type of the boot image.
2. Open a command prompt window and enter the following command:

   WinPE 2.0

   ```
   Createbtimages -i winpe2.2 –o WINPE20 [–c  “c:\program files\Windows AIK”]  [-t path to additional driver]
   ```

   WinPE 3.0

   ```
   Createbtimages -i winpe3.2 –o WINPE30 [–c  “c:\program files\Windows AIK”]  [-t path to additional driver]
   ```

   WinPE 4.0

   ```
   Createbtimages -i winpe4.2 –o WINPE40 [–c  “C:\Program Files\Windows Kits\8.0”] [-t path to additional driver]
   ```

Note: WinPE 4.0 uses Windows Assesment and Development Kit (ADK). As a result, the path should be the ADK install path, for example “C:\Program Files\Windows Kits\8.0”.

OSIM Creating Windows PE 2.0, 3.0, and 4.0 (64-bit) with CreateBtImages

To create a Windows PE 2.0, 3.0, and 4.0 64-bit ready to use with RegisterBtImages -i *name* - s *domain_manager*:

1. Install Windows AIK, or Windows ADK on the IPS box based on the type of the boot image..
2. Open a command prompt window and enter the following command:

   WinPE 2.0 x64

   ```
   Createbtimages –i winpe264.2 –o WINPE20X64 [–c  "c:\program files\Windows AIK"]
   [-t path to additional driver]
   ```

   WinPE 3.0 x64

   ```
   Createbtimages –i winpe364.2 –o WINPE30X64 [–c  "c:\program files\Windows AIK"]
   [-t path to additional driver]
   ```

   WinPE 4.0 x64

   ```
   Createbtimages -i winpe464.2 –o WINPE40X64 [–c  “C:\Program Files\Windows
   Kits\8.0”]  [-t path to additional driver]
   ```

Note: WinPE 4.0 uses Windows Assesment and Development Kit (ADK). As a result, the path should be the ADK install path, for example “C:\Program Files\Windows Kits\8.0”.

CreateBtImages Options for Windows PE

The following CreateBtImages options are specific to Windows PE:

-t <Path to additional drivers>

Specifies the path to additional drivers (*.inf, *.sys) for a WinPE image. For WinPE20, WinPE30, WinPE40, WinPE20x64, WinPE30x64, and WinPE40x64 you can store the driver files in sub directories below the given driver path. The additional driver files are added to the driver directory of the boot image template in the image prepare system. The drivers are then inserted in any new boot image that is based on the same type and template.

Default: Adds the drivers that already exist in the boot image template

-c <Path to WinPE 2005 or Windows AIK or Windows ADK installation>

For WinPE 2005 the path to the Windows PE directory. The winpe directory must contain a copy of the winpe directory from the CD, “Microsoft Windows Preinstallation Environment 2005” (Microsoft Part No. T94-00005).

For WinPE 20 and WinPE30, download the Windows AIK (WAIK) 1.1 or Windows AIK (WAIK) 1.1 DVD respectively from the Microsoft download sites and install Windows AIK from the DVD. The install directory, which is normally “c:\program files\Windows AIK,” must be given here.

Default: Path to the installed Windows AIK

For WinPE 40, download the Windows Assessment and Deployment Kit (ADK) from the Microsoft download sites, and install it on your local Image Prepare System (IPS). The value of this parameter is the Windows ADK install path, for example "C:\Program Files\Windows Kits\8.0".

-s <Temporary destination directory for the WinPE OSIM image>

Temporary directory for files needed to create the WinPE image on the IPS. The directory must be empty.

Default: Directory in image store.

OSIM Tools for Windows PE

For WinPE images, OSIM provides certain tools that you can use to communicate with the boot server and to manage the OSIM parameters. The tools are available in the Image Prepare System templates and are inserted automatically with the CreateBTImages command so that you can modify an OSIM WinPE image with custom actions. This chapter describes these tools.

This section contains the following topics:

osimrun.cmd script

The OS Installation Management starts the script within the created WinPE, finds the boot server, prepares the hard disk, and loads and executes the operating system setup script from the boot server.

Note: This script is customizable.

osimdisk.txt

This file contains the default partition schema of the target's hard disk, and is the input for the Microsoft diskpart partition tool.

For more information, refer to the diskpart documentation.

sdmpcimg Command (32bit and 64bit Version)

Valid on win32

The sdmpcimg.exe command provides options for the WinPE image support. These options include the following:

- Finding the CA boot server the target is assigned to.
- Retrieving all OS Installation Management parameters for this target from the boot server.
- Downloading a file using Trivial File Transfer Protocol (TFTP) from the boot server.

This command has the following format:

```
sdmpcimg [-m tftp] -[lptvz] ...
```

l

Get binl information about local interfaces and boot server.

p

Download parameter file found in binl packet from boot server:

`-p [file/directory to download parameters to] [server ip] [server port]`

Note: If [file/directory to download parameters to] is not provided, then the parameter file is downloaded to the current directory.

t

Get a file from the tftp server:

`-t <file to download> [local file/directory to download to] [server ip] [server port]`

v

Print version information.

z

Download caz file from server:

`-z <caz file to download> [directory to download to] [server ip] [server port]`

If [*file/directory to download to*] is not provided, then the file is downloaded to current directory.

If [*server ip*] is not provided, the IP address from the binl packet is used.

If [*server port*] is not provided, a connection is made to the default tftp port.

Example: Find the Boot Server

This example finds the CA boot server where the target is assigned to:

```
sdmpcimg –l
```

The command provides the following boot server information as standard output:

- DHCP server
- Network mask
- Default router
- Boot server
- Parameter file name

Example: Get All OSIM Parameters for This Target

The following example gets all OSIM parameters from the boot server and writes them in the form $parametername$=value into the parameter.dat file in the local c: directory:

```
sdmpcimg –m TFTP –p c:\parameter.dat
```

The following example gets all OSIM parameters from the boot server and writes them in the form $parametername$=value to standard output:

```
sdmpcimg –m TFTP –p !
```

Example: Download File Using TFTP

This example downloads a file called file01, using TFTP from the boot server, to the target file file02 in the local c: directory:

```
sdmpcimg –m TFTP –t #MPCPATH#/images/dosboot/winxpx86.2/file01 c:\file02
```

The boot server replaces #MPCPATH# with the path to the image store.

Configure TFTP Blocksize to Improve File Transmission Time

By default, the sdmpcimg command requests1456 data bytes per TFTP packet (blocksize) when initiating a TFTP connection. You can configure the requested TFTP blocksize to improve the transmission time of large files. The requested blocksize is defined in an initialization file sdmpcimg.ini which resides in the same folder as the sdmpcimg binary

Note: Applicable only to Windows PE 2 or Windows PE 3 boot images.

To configure TFTP blockize

1. Open the initialization file smdpcimg.ini located in the following folder in the Image Prepare System server.:
   - 32-bit: *ITCM_installpath*\osimips\os-template\updates\winpe20\i386\ca-osim
   - 64-bit: *ITCM_installpath*\osimips\os-template\updates\winpe20x64\amd64\ca-osim.
2. Increase the blocksize value. Acceptable numbers are from 8 to 65464.

   Increasing the number significantly reduces the transmission time of large files.
3. Save the file.
4. Create a boot image or update an existing boot image. For more information about updating boot images, see Update Boot Images (see page 21).

   The updated initialization file is included in the boot image. By default, the boot server accepts the TFTP blocksize that sdmpcimg requests. However, you can configure the boot server to limit the effective TFTP block size. To do this, follow these steps :

   a. Open configuration policy assigned to the boot server in DSM Explorer.

   b. Navigate to DSM, Scalability Server, OSIM, ManagedPC, Server, and reduce the TFTP specific blocksize limit policy as per the value specified in the smdpcimg.ini file.

      The policy limits the number of TFTP data bytes per packet for connections initialized by sdmpcimg

      Note: PXE firmware, DOS boot images and Client Automation r11.x or r12 Windows PE boot images are not affected by the parameter TFTP specific blocksize limit. The parameter TFTP blocksize limit can be used to limit the blocksize used with these TFTP clients, see PXE-32: TFTP Open Time-out. DOS boot images and Client Automation r11.x or r12 Windows PE boot images do not support a TFTP blocksize < 1024.

preplace Command--Substitute OSIM Parameters in Text Mode Files

Use the preplace.exe command to substitute OSIM parameters in text mode files. This command also applies to WinPE images.

Note: There are different versions for win32 and amd64.

This command has the following format:

preplace -i *parameter_file*$_1$ [... -i *parameter_file*$_n$] *file*$_1$ [... *file*$_m$]

where:

-i *parameter_file*$_1$ [... -i *parameter_file*$_n$]

Specifies a list of files containing the OSIM parameters. The list elements are separated by spaces.

***file*$_1$ [... *file*$_n$]**

Specifies a list of files in which the parameters should be substituted. The files in the list are separated by spaces. In all files listed, each parameter value of $*parametername*$ is replaced with the value from the file *parameter_file*.

Example: Substitute OSIM Parameters

The following command sequence gets the Windows XP installation script winxp.cmd (see page 145) and auto answer file winxp.inf, as well as the target specific parameters from the boot server. The preplace command substitutes the OSIM parameters in both the installation script and the auto answer file, before the Windows XP installation script is executed.

```
sdmpcimg –m TFTP –t #MPCPATH#/camenu/winxp.inf c:\winxp.inf
sdmpcimg –m TFTP –t #MPCPATH#/camenu/winxp.cmd c:\winxp.cmd
sdmpcimg –m TFTP –p c:\parameter.dat
preplace –i c:\parameter.dat c:\winxp.inf c:\winxp.cmd
C:\winxp.cmd
```

canet Command-Get Access to OSIM Shares on the Boot Server

Use the canet.exe command to do the following:

- Encrypt a user's password.
- Change a user's password.
- Change the administrator password.
- Add workstation to a domain.
- Connect to a shared disk resource on a boot server as user canonprv.

Note: There are different versions for win32 and amd64.

This command has the following formats:

- Encrypt a user's password:

  `canet crypt [ user ] password [ /X ]`

- Change a user's password:

  `canet user user encrypted_password [ /PWExpires:{Yes|No} ]`

- Change the administrator password

  `canet administrator encrypted_password [ /PWExpires:{Yes|No} ]`

- Add workstation to a domain:

  `canet JoinDomain domain [ MachineObjectOU=full_LDAP_path_of_OU> ] [ user [ encrypted_password ]]`

- Connect to a shared disk resource on a boot server as user canonprv:

  `canet use [device] \\computer\share [ encrypted_password [ /USER:[domain\]canonprv ] ]`

Example: Use canet to Get Access to the Operating System Image on a Boot Server Share

This excerpt from a sample Windows XP installation script, winxp.cmd, shows how to use canet to get access to an operating system image on a boot server:

```
...
canet use s: \\$BootServer$\winxp $LogonPasswd$ /USER:$BootServer$\canonprv
s:
cd \
i386\winnt32.exe /tempdrive:c /syspart:c: /noreboot /unattended:c:\winxp.inf
...
```

Before you execute this installation script from WinPE, substitute the OSIM parameters (see page 144).

decodDat.exe Command--Create a Parameter File With Localized Parameter Values

This decodDat command creates a parameter file with localized parameter values. It searches the input files for parameters with names that end with "_UTF8$". When found, their values are converted to the given locale. "_UTF8" is removed from their names and the resulting parameter is written to the output file. Non-convertible characters are replaced by "?".

Note: There are different versions for win32 and amd64.

This command has the following format:

```
decoddat [-v] [-l locale] [-d charmaps_directory] [-u] -o output_file input_file
```

-v

If present, the copyright and version information is written to stdout.

-l *locale*

Name of the locale.

Default: ENU

-d *charmaps_directory.*

Name of the charmaps directory.

Default: CAWIN

-u

If present, a parameter $OSPasswdUxSV$ is generated from $OSPasswd$.

-o *output_file input_file*

Name of the output file and the input file.

Example

```
decodDat -l ENU -d m:\DOSmaps -o c:\canpcloc.dat c:canpcext.dat
```

Internal Boot Server Structure of WinPE Boot Images

The boot image directory, UNDI, can include DOS boot images like osinstal.2 and osinstal.3 or description files of boot images, like winpe.

Nevertheless, a sequence of multiple boot images is possible even in combination with DOS boot images. E.G. winpe2.2, winpe2.3, winpe2.4 , boothd.

A description file defines a link to the real WinPE2.2 boot image directory, which can contain multiple files.

Example: Contents of the Description File for winpex86.2

```
[BootImage]

;Properties of the image

imageflags=F140-2 petftp
;Type of bootimage
Boottype=WinPE2
;Directory containing the bootimage files
Bootimagedir=winpe2.2
;Network loader for the bootimage which is stored in "Bootimagedir"
Bootloader=pxeboot.n12
;The boot server switches to the next boot image or boothd
;if this file from "Bootimagedir" is downloaded.
Switchfile=winpe.wim
```

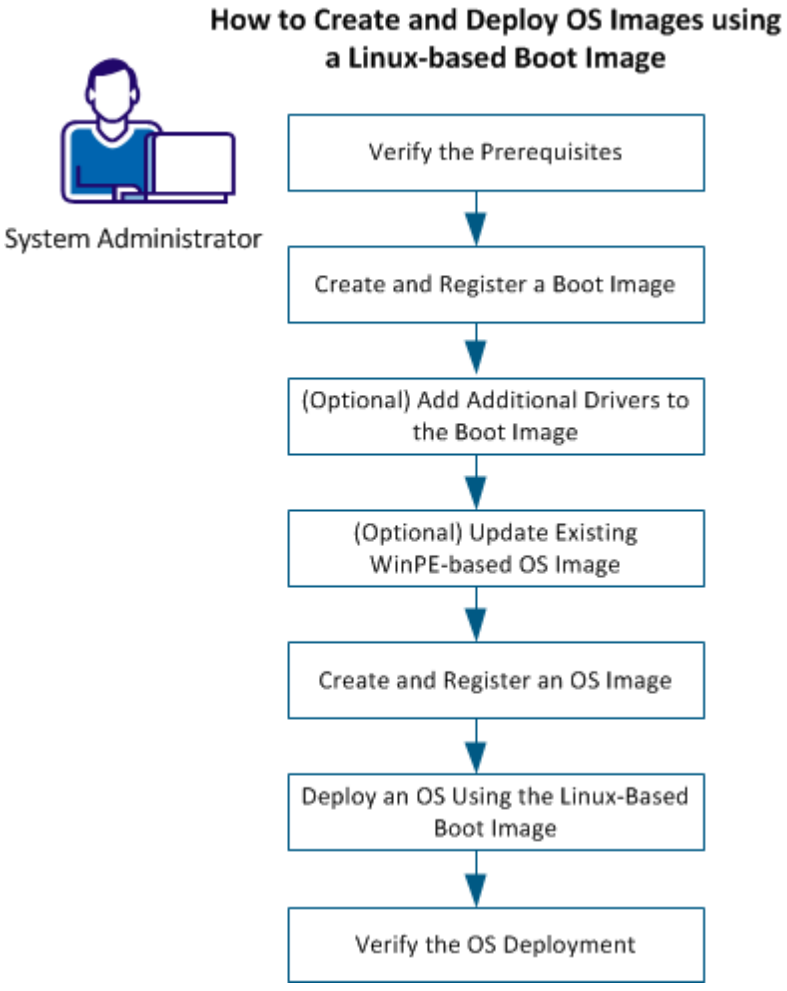
Enhanced Flexibility with WinPE Boot Images

Microsoft provides various versions of Windows PE. Check the Microsoft documentation to find the correct Windows PE version for your operating system.

Chapter 8: How to Create and Deploy OS Images using a Linux-Based Boot Image

As a system administrator, you can use a Linux-based boot image to deploy Linux operating systems and Hypervisors like VMware ESXi and Citrix Xen Server in an unattended mode.

The following diagram describes the steps for creating and deploying the OS images using Linux-based boot image:



Using the new Linux-based boot image, you create and deploy OS images. See Supported Operating Systems for the list of Operating Systems supported.

Perform the following steps:

1. Verify the Prerequisites (see page 150)
2. Create and Register a Boot Image (see page 153)
3. (Optional) Add Additional Drivers to the Boot Image (see page 152)
4. (Optional) Update an Existing OS Image (see page 153)
5. Create and Register an OS Image (see page 156)
6. Deploy an OS Using the Linux-Based Boot Image (see page 157)
7. Verify the OS Deployment (see page 158)

This section contains the following topics:

Verify the Prerequisites

Client Automation provides a new Linux-based boot image that is based on CentOS 6.3 64-bit OS. Verify you have access to the media with a customized version of this Linux-based boot image.

Note: You can create, update, and register the boot and OS images using the corresponding Image Prepare System wizards in DSM Explorer or the CLI. You need administrator privileges on the IPS installed computers.

On the computers with pre-Windows Vista, such as Windows XP, Windows Server 2003, or lesser versions, the user must be a part of the Local Administrators Group.

On Windows Vista and above such as Windows 7, Windows Server 2008, and higher, where the user is not the built-in administrator, the user must be part of the local administrators group, and run the tools using *Run as administrator* in an elevated mode.

Create and Register a Boot Image

You can create a Boot image by using the corresponding wizards in DSM Explorer or the corresponding commands from the Command Line Interface (CLI).

For more information, see Create and Register a Boot Image using DSM Explorer (see page 151) and Create and Register a Boot Image using CLI (see page 151).

Create and Register a Boot Image using DSM Explorer

Follow these steps:

1. Navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, New Boot Image.

   The wizard displays the introduction content. Click Next.
2. Select the LINUXPEx64 option from the list of Image types. Click Next.
3. Enter a name for the boot image in the Image Name field. Click Next.
4. Browse and select the folder that contains the customized Centos 6.3 x64 boot files, which are provided on a DVD separate from the product DVD. Click Next.
5. Review the summary of the details you have specified. Click Finish.

   The New Boot Image: The Command Execution page shows that the boot image is successfully created.

You can now register the boot image.

Follow these steps:

1. Verify that the new boot image is available under Software, Boot and OS Image Library, Image Prepare System, Local Boot Images.
2. Right click the boot image and click Register Boot Image in Connected Domain, Data and Software Package.

   Register Boot Image: The Command Execution page shows that the boot image is successfully registered.
3. Verify that the boot image is available in the Boot Images folder of the Boot and OS Image Library.

You can proceed to create and register the OS image.

Create and Register a Boot Image using CLI

You can create and register a boot image using CLI.

Follow these steps:

Execute one of the following commands to create a boot image:

- Use the following command if the DVD with the customized CentOS 6.3 x64 boot image, is inserted on the Image Prepare System:
  `CreateBTImages -i <Image Name> -o LINUXPEx64`
- Use the following command if the DVD content is extracted to a folder on the Image Prepare System:
  `CreateBTImages -i <Image Name> -o LINUXPEx64 -m <Path to the Customized CentOS image>`

  Where –m specifies the location of the folder where the extracted customized CentOS boot files exist. For example, the DVD is extracted to C: drive, the path is *C:\linuxpe\x86_64*

Follow these steps:

Use the following command to register the boot image:
`RegisterBTImages {-s <manager>} [-i <imagenames>[-u <user> -p <password> -d <domain>]`

(Optional) Add Additional Drivers to the Boot Image

Add drivers to the boot image when the Linux boot image does not contain the drivers that are required for the target hardware. A Linux-based machine is required for the same. CentOS 6.3x64 is the preferred OS.

Follow these steps:

1. Navigate to the customized Linux boot image files available on a boot image DVD. Copy the initrd.img (RAMDISK image) and unpack it by using the following commands:

   ```
   # mkdir initrd

   # cd initrd

   # gzip –dc ../initrd.img | cpio –idv
   ```

2. Copy .ko files to the appropriate directory under /lib/modules/2.6.../kernel/drivers/

   When additional drivers (for example: *myscsidriver.ko*) are added for SCSI disk, copy the myscsidriver.ko files to the appropriate /lib/modules/ 2.6.32-279.el6.x86_64/kernel/drivers/scsi directory of the initrd image.

   ```
   # cp myscsidriver.ko  /lib/modules/2.6.../kernel/drivers/scsi/
   ```

3. Create a backup of the existing initrd.img.

   ```
   #mv  ../initrd.img  ../initrd.bak
   ```

4. Create new initrd.img by repacking files and directories into the initrd.img it by using the following commands:

   ```
   # find.| cpio -H newc -o > ../initrd.cpio

   #gzip ../initrd.cpio

   #mv../initrd.cpio.gz ../initrd.img
   ```

Note: When you create initrd.img, use a separate directory to create initrd.cpio and initrd.img. If you use the same directory that contains all the unpacked files, cpio attempts to create initrd.cpio recursively (packing again and again), and may fail due to lack of sufficient space on the hard disk.

(Optional) Update Existing WinPE-based Linux OS Images

You can update the existing Linux OS images that use WinPE as the boot image to use the new Linux-based boot image for deployment. See the Compatibility Matrix available on support.ca.com for the list of Operating Systems supported.

Use the following methods for updating the OS images.

- Update OS image Image prepare system wizard or
- CreateOSImage/RegisterOSImage utilities.

This updates the existing OS images with new scripts and utilities needed to work with the new Linux-based boot image and sets up the OS image to use this new boot image.

With the new boot image support, administrators do not have to copy syslinux manually in the OS Image.

Update OS Images using DSM Explorer

Update the existing WinPE-based Linux OS images with the new template files to be able to use the new Linux-based boot image.

Follow these steps:

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Navigate to Software, Boot and OS Image Library, Image Prepare System, Update OS Image in the DSM Explorer.

   The Update OS Image wizard appears.
3. Follow the instructions in the wizard to update the OS image.

   The OS image is updated and registered with the domain manager based on the options you have selected. Deploy the updated SD package on the other boot servers that stage the OS image that you just updated.

More information:

Make Boot and OS Images Available on Boot Servers (see page 52)
Updating Customized Files (see page 18)

Update OS Images using CLI

Use the command line if you are updating multiple OS images using a batch script.

Note: If you have deployed the same OS image on multiple boot servers, update the image and create a corresponding update SD package on the Image Prepare System first. You can then deploy the updated SD package to all the other boot servers in the domain using a software delivery job.

Follow these steps:

1. (Optional) Verify that you have merged the custom changes, if any, to the new template files.
2. Execute the following command:

   ```
   CreatOSImage -i <imagename> -e
   ```

   The command updates the existing image with the new templates.
3. Execute the following command to update the OS image on the domain manager:

   ```
   registerOSImage –i <image name> -s <manager> -e
   ```

   The command updates the image on the domain manager specified with the -s parameter.

4. Execute the following command if you need an SD package for updating the OS image on a remote boot server:

   ```
   registerOSImage –i <image name> -s <manager> -e –l
   ```

   The command creates an SD package with the updated OS image. The updated package has "-update" appended to the original package name. The updated package can be used to update the OS image on remote boot servers using an SD job with the *Update Image on Boot Server* procedure.

   Execute the following command to update the parameter definitions of the image:

   Important! You may lose the customized parameter definitions if they have been updated or deleted in the new template.

   ```
   registerOSImage –i <image name> -s <manager> -e –b
   ```

   The command updates the parameter definitions of the OS image in the MDB. The updates include changes to the default values, new, updated, or deleted parameter definitions.

The OS image is now updated in the domain manager and in the boot servers.

More information:

Make Boot and OS Images Available on Boot Servers (see page 52)
Updating Customized Files (see page 18)

Create and Register an OS Image

You can create an OS image by using the corresponding wizards in DSM Explorer or the corresponding commands from the CLI.

Create and Register an OS Image using DSM Explorer

Follow these steps:

1. Verify that the required OS DVD is inserted and readable in a local optical device (CD or DVD).
2. Select New OS Image from the Image Prepare System Wizards folder. Click Next.
3. Select the required OS from the list of operating systems. Click Next.
4. Enter a name for the OS image. Click Next.
5. Click Finish to initiate the copy of the installation files into the Image Prepare System and boot server store.

   The image is created and is available to the Image Prepare System.

   You can now register the OS image.

Follow these steps:

1. Open the DSM Explorer and navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, Register OS Image.
2. Follow the instructions in the wizard and register the OS image with the domain.

   Note: When registering, select the option to stage the SD deployment image for distribution to one or more boot servers using Software Delivery.

Create and Register an OS Image using CLI

Create an OS Image:

Use the following command to create a new OS Image:

```
createosimage -i <image Name> -o <OS Type> -s <OSpath>
```

The following example creates a new RedHat x64 OS image RH55x64 and reads the installation files from a folder.

```
createosimage -i RH55x64 -o RH5XPE -s f:\redhat_55_x65_files
```

Register this image with the DSM domain to be made available for deployment. Use the OS image registration wizard or the command line to register the image. When registering, you can opt to create an SD deployment image for distribution to one or more boot-servers using Software Delivery.

Register the OS Image:

Use the following command to register the boot image:

```
RegisterOSImage -s <manager> -i <imagename> | -w <directory> [-b] [-l] [-t] [-e] [-n <name>]
[-v <version>] [-c <comment>] [-u <user> -p <password> -d <domain>]
```

Deploy an OS Using the Linux-Based Boot Image

To perform an unattended installation of the OS, activate the OS deployment on the target computer.

Follow these steps:

1. Navigate to DSM Explorer, Computers and Users, All Computers and find the target computer on which you want to deploy the OS.
2. Right-click the computer and select OS Installation.

   The OS Installation wizard opens.
3. Follow the instructions in the wizard and click Finish on the last page.

   An OS deployment job is created and the deployment status is displayed in the Computers and Users, All Computers, Group Details, and OS Installations node.

OS is deployed using the Linux-based boot image.

Note: The disk (either local or remote) being used for OS deployment must be the first disk in the boot sequence among all the available bootable disks (disks holding any bootable OS or bootable media).

Client Automation supports the installation of RHEL 5.x and 6.x OS on the first local disk only and this disk must be the first disk in the BIOS boot sequence.

Deploy an OS to a SAN Target

In a SAN environment, this task involves assigning a Disk Identifier (SANID) to the OS image so that it can identify the SAN disk on which deployment is planned.

Understand the SANID Boot Parameter

The SANID boot parameter is added to the OS images to identify a specific target related to a particular SAN logical unit number (LUN) or Disk. This provision ensures that the specified LUN or Disk is explicitly selected. SANID boot parameter holds the identifier value that is associated with the specified Disk or LUN like World Wide Name (WWN) or World Wide Identifier (WWID) based on Network Address Authority (NAA) format.

The supported formats (with example values) for SANID are:

```
value
    naa.60a9800064762f34694a6e3651697759
    60a9800064762f34694a6e3651697759
```

For backward compatibility, support for the SANID value in the *attribute=value* format is available as shown below:

```
    TargetLUN=naa.60a9800064762f34694a6e3651697759
```

The support for providing the SANID parameter using EUI format is dropped from this release.

When the SANID value is empty, the installation scripts use the first available local disk for the installation per the existing behavior. Also, this disk must be the first disk in the BIOS boot sequence.

Verify the OS Deployment

After the OS deployment is started, system administrator can verify the status of the job from the job status.

Parameters that influence the Job Status:

The job status depends on the following two parameters:

- InstallAgent
- SignalCompletion

SignalCompletion signals the completion of the OS installation in addition to the InstallAgent parameter.

Client Automation agent installation is not supported on VMware ESXi 4.1, VMware ESXi 5.1, and Citrix XenServer hypervisors. Hence, InstallAgent parameter is set to No by default for these operating systems.

The SignalCompletion and InstallAgent parameters have the following use cases:

InstallAgent Value	SignalCompletion Value	Status of the OS Installation Job
Yes	Yes	Job is marked as *current* after successful OS installation.
Yes	No	Job is marked as *current* on the OS that have agent support in Client Automation. For other OS like hypervisors, the job remains in *installing* state.
No	Yes	Job is marked as *current* after the successful OS installation.

No	No	Job is marked as *current* after the Boot Image downloaded.

Logging and Troubleshooting

The installation process of OS writes some files back to the boot server which possibly helps to detect installation problems. The files are available in the OSIM-targets folder in the used OS image of the image store on the boot server. A folder representing the target computers MAC address contains these files:

canpc.dat

Lists all boot parameters that are defined for the OS installation process.

setup.bat

Defines the boot parameter replaced script that is executed in the Linux PE boot phase.

ks.cfg /xens61.xml/autoinst.xml/preseed.cfg

Defines the boot parameters (auto answer file for unattained installation) which are replaced by Linux PE boot phase and used by native OS installation.

Ks.cfg for RHEL and ESXi.

xen<OSVersion>.xml like xen61.xml for XenServer.

autoinst.xml for SUSE and OpenSUSE.

Preseed.cfg for Kubuntu

preinst.log

Contains a log of the OSIM pre-installation process (Linux PE boot phase) on the target computer.

postXinst.log

Contains a log of the OSIM post-installation process (OS customization, Agent Installation, and Signaling) on the target computer.

syslinux.cfg/ boot.cfg

Indicates the boot-loader configuration file to specify the location of the auto-answer file for the unattained installation.

custom.cmd/xensfail.sh/xenspost.sh

Indicates post-install script to execute for OS customization Agent Installation and Signaling.

Chapter 9: Windows OSIM OS Images

This section contains the following topics:

SAN Deployment Support for Windows Server

SAN Deployment Supports for Windows Server 2012 and Windows Server 2012R2.

Windows Server OS can be deployed in a SAN environment by using OSIM. Target SAN disk for OS deployment is identified by WWN or WWID based on NAA addressing scheme.

Boot parameter SAN ID is available to specify WWN or WWID of Disk. The SANID boot parameter is added to the Windows Server OS images to identify a specific target related to a particular SAN logical unit number (LUN) or Disk. This provision ensures that the specified LUN or Disk is explicitly selected. SANID boot parameter holds the identifier value that is associated with the specified Disk or LUN like World Wide Name (WWN) or World Wide Identifier (WWID) based on Network Address Authority (NAA) format.

The supported formats, with example values, for SANID are as follows:

naa.60a9800064762f34694a6e3651697759

60a9800064762f34694a6e3651697759

For backward compatibility, support for the SANID value in the attribute=value format is available as follows:

TargetLUN=naa.60a9800064762f34694a6e3651697759

SANID boot parameter is ignored if boot parameter “CreateNewPartitions” is set to true because, in this case it uses any existing available partition (C drive) created by previous windows installation for new OS Installation.

Supplying boot parameter SANID with boot parameter “WinPEPartition” with value “Use special partition file W2012.wp1 from CAMENU”, forces OSIM to ignore parameter “WinPEPartition” value.

If SANID is specified, OSIM identifies the disk to be used for OS installation and the supplied external partition file is ignored for OS deployment.

For more information, see Chapter 2 “How to Use OSIM”, Section Manage OS Installations “Boot Parameters”.

Windows Key-less Installation

Key-less Installation is supported for Windows 8, 8.1 Server 2012, and 2012 R2.

Windows can be deployed without specifying the Product Key through OSIM. Later, it can be activated by using either supplying valid Product key, Volume License Key or by using Volume Licensing Server.

In an Enterprise, customer may deploy multiple Windows OS with leave Product ID or Key Field empty and later these can be activated by using Volume Licensing Server.

This feature is specific to particular Editions like in case of Windows 8 or 8.1, Enterprise and ProVL edition can be installed without the supplied Product Key and windows can be activated later by using Volume License mechanism or specifying the valid Product Key. Key-less installation does not hold true for all Windows OS Editions. For information, refer to MSDN or Windows Installation documentations.

Volume Licensing (VL) Image Installation for Windows

Volume Licensing (VL) Image Installation is supported for Windows 8, 8.1 Professional Edition.

Windows OS Edition with Volume Licensing Image can be deployed through OSIM.

Windows releases image type such as “Windows 8.1 Pro VL (x64)" which is Volume Licensing Image of Windows 8.1 Pro 64-bit Edition. To use VL image of Windows OS Edition, check Boot parameter "WindowsType" value.

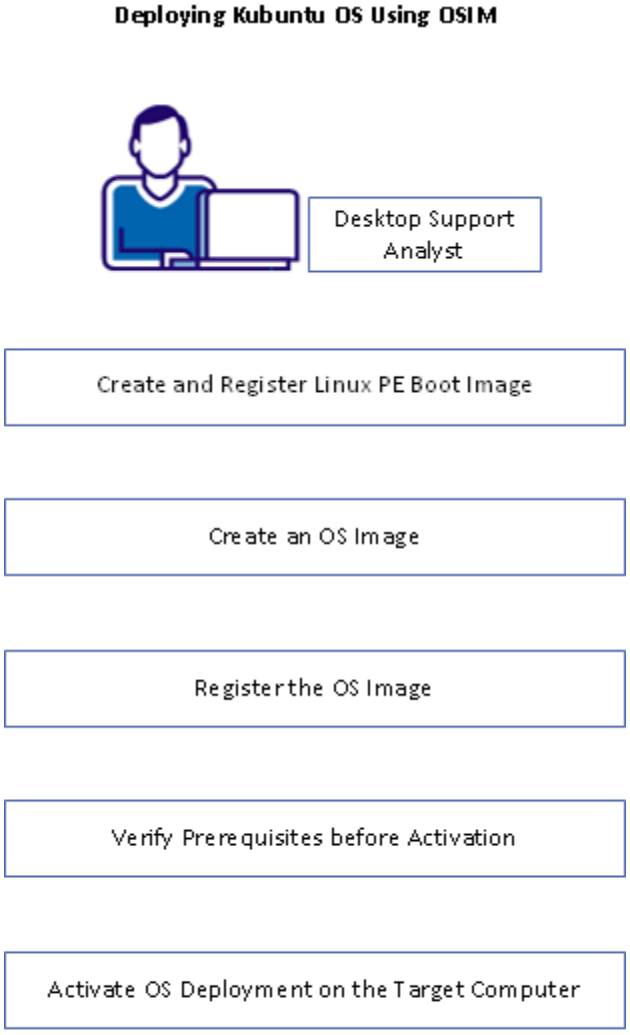
Chapter 10: Support for Kubuntu OS and Debian Repositories

This section contains the following topics:

Deploying Kubuntu OS Using OSIM

Important! Kubuntu OS Images are supported with Linux-Based Boot Image only. Recommends using Linux-Based Boot Image instead of DOSX boot image.

Client Automation lets you implement the unattended installation of Kubuntu operating systems on target computers using the OSIM functionality. The following diagram illustrates the steps that you perform to deploy Kubuntu OS using OSIM:



Perform the following tasks to deploy Kubuntu OS using OSIM:

1. Create and Register LinuxPE Boot Image (see page 167)
2. Create an OS Image (see page 169)
3. Register the OS Image (see page 172)
4. Verify Prerequisites before Activation (see page 173)
5. Activate OS Deployment on the Target Computer (see page 173)

Create and Register LinuxPE Boot Image

Before you can deploy an OS, you must boot the target computer using the boot image. You must have supported boot images for each OS. For a complete list of boot images for each supported OS, see the Compatibility Matrix.

Create and Register a Boot Image

You can create a Boot image by using the corresponding wizards in DSM Explorer or the corresponding commands from the Command Line Interface (CLI).

For more information, see Create and Register a Boot Image using DSM Explorer (see page 151) and Create and Register a Boot Image using CLI (see page 151).

Create and Register a Boot Image using DSM Explorer

Follow these steps:

1. Navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, New Boot Image.

   The wizard displays the introduction content. Click Next.
2. Select the LINUXPEx64 option from the list of Image types. Click Next.
3. Enter a name for the boot image in the Image Name field. Click Next.
4. Browse and select the folder that contains the customized Centos 6.3 x64 boot files, which are provided on a DVD separate from the product DVD. Click Next.
5. Review the summary of the details you have specified. Click Finish.

   The New Boot Image: The Command Execution page shows that the boot image is successfully created.

You can now register the boot image.

Follow these steps:

1. Verify that the new boot image is available under Software, Boot and OS Image Library, Image Prepare System, Local Boot Images.
2. Right click the boot image and click Register Boot Image in Connected Domain, Data and Software Package.

   Register Boot Image: The Command Execution page shows that the boot image is successfully registered.
3. Verify that the boot image is available in the Boot Images folder of the Boot and OS Image Library.

You can proceed to create and register the OS image.

Create and Register a Boot Image using CLI

You can create and register a boot image using CLI.

Follow these steps:

Execute one of the following commands to create a boot image:

- Use the following command if the DVD with the customized CentOS 6.3 x64 boot image, is inserted on the Image Prepare System:
  `CreateBTImages -i <Image Name> -o LINUXPEx64`
- Use the following command if the DVD content is extracted to a folder on the Image Prepare System:
  `CreateBTImages -i <Image Name> -o LINUXPEx64 -m <Path to the Customized CentOS image>`

  Where –m specifies the location of the folder where the extracted customized CentOS boot files exist. For example, the DVD is extracted to C: drive, the path is *C:\linuxpe\x86_64*

Follow these steps:

Use the following command to register the boot image:

`RegisterBTImages {-s <manager>} [-i <imagenames>[-u <user> -p <password> -d <domain>]`

Create an OS Image

An OS image includes the operating system installation files that are required for the unattended installation of the OS at the target computers. Create an OS image for the OS that you want to deploy using OSIM. You can create OS images using one of the following methods:

- Create an OS image using the wizard
- Create an OS image using the command line

Verify Prerequisites

- Verify that the Ubuntu media is accessible to the Image Prepare System when creating an OS image.
- Verify that the Ubuntu Alternate CD image is used, and not the Live CD.

Create an OS Image Using the Wizard

You can create an operating system image using the New OS Image Wizard.

Follow these steps:

1. Open DSM Explorer and navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, New OS Image.

   The Introduction page appears.

2. Click Next.

   The Operating System page appears.

3. Select one of the following image types depending upon the version of Kubuntu you want to use and click Next:

   - Kubuntu 10.04 x86 LTS(CD) is for KUBUNTU-10.04-x86.
   - Kubuntu 12.04 x64 LTS(CD) is for KUBUNTU-12.04-x64.
   - Kubuntu 12.04 x86 LTS(CD) is for KUBUNTU-12.04-x86.

   The Image Name page appears.

4. Specify a name for the OS image and click Next.

   The Source page appears.

5. Specify the location of Kubuntu installation files by doing the following tasks:

   a. Select the OS installation files from the OS image files on local or network drive or the CD/DVD Drive.

   b. Click Next.

6. Follow the instructions in the wizard and click Finish.

   The Kubuntu installation files are copied onto the IPS and the boot server store is initiated. When the files are copied, the status is displayed on the last page of the wizard.

Create and Register a Boot Image using DSM Explorer

Follow these steps:

1. Navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, New Boot Image.

   The wizard displays the introduction content. Click Next.

2. Select the LINUXPEx64 option from the list of Image types. Click Next.
3. Enter a name for the boot image in the Image Name field. Click Next.
4. Browse and select the folder that contains the customized Centos 6.3 x64 boot files, which are provided on a DVD separate from the product DVD. Click Next.
5. Review the summary of the details you have specified. Click Finish.

   The New Boot Image: The Command Execution page shows that the boot image is successfully created.

You can now register the boot image.

Follow these steps:

1. Verify that the new boot image is available under Software, Boot and OS Image Library, Image Prepare System, Local Boot Images.
2. Right click the boot image and click Register Boot Image in Connected Domain, Data and Software Package.

   Register Boot Image: The Command Execution page shows that the boot image is successfully registered.

3. Verify that the boot image is available in the Boot Images folder of the Boot and OS Image Library.

You can proceed to create and register the OS image.

Create an OS Image Using Command Line

The OS image creation through the command line is more flexible than the OS image creation through the wizard. For example, you can specify an alternative location for the installation files.

Example: Command for Kubuntu OS Images

The following example creates a Kubuntu OS image that is named kubuntu1 with an OS type as KUBUNTU-10.04-x86:

```
createosimage –i kubuntu1 –o KUBUNTU-10.04-x86
```

Register the OS Image

After the image is created locally on the IPS, register the OS image with the DSM domain to make it available for deployment.

You can register OS images using one of the following methods:

- Register an OS image using the wizard (see page 172)
- Register an OS image using the command line (see page 172)

Register the OS Image Using the Wizard

Follow these steps:

1. Open the DSM Explorer and navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, Register OS Image.
2. Follow the instructions in the wizard and register the OS image with the domain.

   Note: When registering, select the option to stage the SD deployment image for distribution to one or more boot servers using Software Delivery.

Register the OS Image Using Command Line

Example: Command to register an OS image

The following example registers an OS image from the Image Prepare System with the associated domain manager:

```
registerOSImage -s managername -i imagename
```

Note: For more information about the commands, parameters, and examples, see the OS Installation Management Administration Guide.

Verify Prerequisites before Activation

Verify the following prerequisites before activating OS installation:

- Verify that the boot and repository servers are configured for HTTP and FTP access. For more information, see Setting Up HTTP and FTP Share for Software Packages and OS Images.
- Verify that the system clock on the target computer is set to UTC, if you want the OS time to be in a specific time zone. If the system clock is based on the local time, the OS time is also set to local time during installation.

  Note: Alternatively, you can change the time setting in the preseed file before deploying the OS image. Locate the d-i clock-setup/utc boolean true setting and change true to false. You can find the preseed file at the following location:

  DSM Installation Directory\Server\SDBS\var\managedpc\images*Image Name**Image Name*\IMAGESUP

  Example:

  E:\CA\DSM\Server\SDBS\var\managedpc\images\K1204x64\K1204x64\IMAGESUP

Activate OS Deployment on the Target Computer

To perform an unattended installation of the OS, activate the OS deployment on the target computer.

Follow these steps:

1. Open DSM Explorer and navigate to Computers and Users, All Computers and find the target computer on which you want to deploy the OS.
2. Right-click the computer and select OS Installation.

   The OS Installation wizard opens.
3. Follow the instructions in the wizard and click Finish on the last page.

   An OS deployment job is created and the deployment status is displayed in the Computers and Users, All Computers, Group Details, and OS Installations node.

Kubuntu OS Installation Limitations and Issues

The following notes and issues apply for OS installation on Kubuntu using OSIM:

- Shortly after the installer starts executing on the target computer, the console screen will stop showing any progress and shows a solid blue screen. This stays like this for several minutes before the installation continues to display progress. The netboot installer appears to be processing the downloaded file information to build up a final download list which includes numerous processing of files and MD5 hashes. When run in a VM environment, you can see that the processor is highly utilized in this phase. For a physical computer, wait for the installer to resume as nothing can be done during this phase. To verify that the computer has not locked up (hardware type fault), use the Alt+Fn keys to switch between virtual consoles; the installer runs on Alt+F1 and the syslog output is shown on the console Alt+F4. Alt+F2 and F3 allow for an interactive shell during installation, but there are no process monitoring tools, such as top, available in this environment.
- Verify that the alternate CD is used when the identfile is not detected on the inserted CD.
- The Ubuntu Live CD cannot be used for OSIM as a netboot capable kernel and initramfs are required which is only provided on the alternate and netboot ISO.
- Deployment of Kubuntu requires manual configuration of FTP/HTTP servers.
- OSIM allows the default username to be changed to "root" (or any other system reserved names such as wheel). The installation fails if system reserved names are specified.

OS Installation Verification, Logging, and Troubleshooting

For information about verifying the OS deployment, see the Verify the OS Deployment (see page 158). For information about OS installation logging and troubleshooting, see the Logging and Troubleshooting (see page 159).

Setting Up FTP and HTTP Share for Software Packages and OS Images

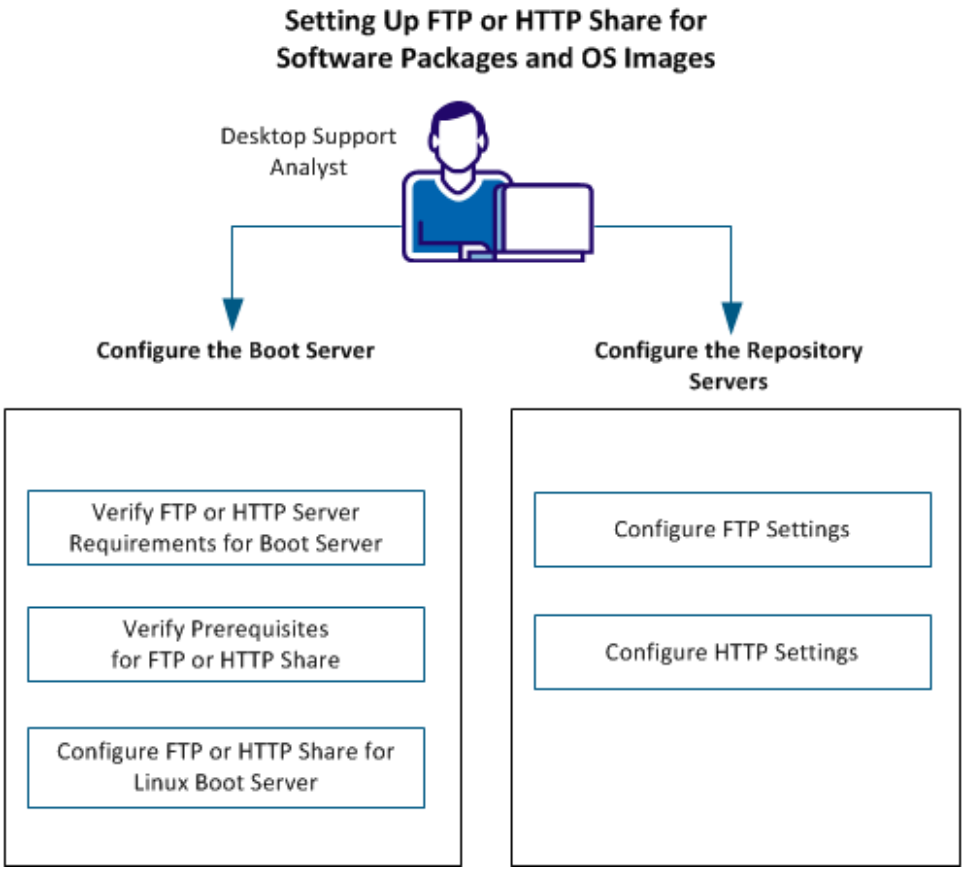
As a desktop support analyst, you are required to set up FTP or HTTP share to store software packages and OS images. Expose the location that contains packages and OS images as FTP or HTTP share on the following servers:

- For OS images, configure the boot server and repository servers (master and mirror).

  Note: If the boot and repository servers co-exist on the same computer, configure only one computer.

- For software packages, configure only the repository servers.

The following diagram illustrates the steps that you perform to set up FTP or HTTP share for software packages and OS images:



Perform the following tasks to set up FTP or HTTP share on boot servers:

1. Verify FTP and HTTP Server Requirements for Boot Server (see page 176)
2. Verify Prerequisites for HTTP Share (see page 177)
3. Configure FTP and HTTP Shares for Linux Boot Server (see page 178)

Perform the following tasks to set up FTP and HTTP share on repository servers:

1. Configure FTP Settings (see page 179)
2. Configure HTTP Settings (see page 180)

Configure the Boot Server

Before you deploy the OS image, you must have configured the boot server for HTTP or FTP share.

Verify FTP or HTTP Server Requirements for Boot Server

The boot server must have the following configuration for HTTP or FTP shares, depending on their operating environment.

Note: For more information about setting up HTTP or FTP shares, see the operating system documentation.

Windows:

- IIS 6 and IIS 7.5 for HTTP share
- FTP 6 and FTP 7.5, but not FTP 7.0, as FTP 7.0 has no management tools except for WMI.
- IIS 7.5 for FTP

  Note: Windows 2008 ships with IIS 7.0 and FTP 7.0. As FTP 7.0 is not supported, upgrade to FTP 7.5 by upgrading IIS to 7.5.

Note: Verify that the HTTP or FTP server on Windows are configured to have a default website. OSIM uses the default website to expose virtual directories for storing configuration and netboot files for OS installation.

Linux:

- Any HTTP server that serves a folder hierarchy that is mounted through a symbolic link. For example, Apache 2.
- Any FTP server that serves a folder hierarchy that is mounted through a file system bind point. For example, vsftpd.

Note: The configuration files for the OS image installation are read from the HTTP or FTP servers that are located with the boot server, even if the repository is configured to use a central mirror.

Verify Prerequisites for FTP or HTTP Share

Verify the following prerequisites before you configure the FTP or HTTP share on Microsoft IIS or standard Linux HTTP server type.

Configure HTTP Share on Microsoft IIS

- Verify that the servers are configured to support the correct MIME types for all packages in the mirror as IIS only supports known file types.
- Verify that files in the URL are accessible by Browser. Try accessing a file in the default web site using the browser. For example, http://*hostname*/test.txt.

Configure FTP Share on Microsoft IIS

- Verify that anonymous access is allowed on the default FTP site. Using an FTP client, log in as an anonymous user using an email address as the password and verify that the top-level folder is accessible.

Linux HTTP Server

- Verify that the http server is running on the boot server. You can check the status by using following command.

  ```
  #ps –ef | grep http*
  ```

- Verify that files under the Http Document Root are accessible by Browser URL. Try accessing a file in the Http Document Root using the browser. For example, http://*hostname*/test.txt.

- If you are not able to access the symbolic link under Http Document Root using the browser, verify the http configuration for disabled symbolic link configuration. The following example shows how you can edit or add lines in the http configuration file for Apache:

  ```
  <Directory "$Document_Root">
  Options Indexes MultiViews FollowSymLinks
  AllowOverride None
  Order allow,deny
  Allow from all
  </Directory>
  ```

  Note: For more information about the configuration file for your HTTP server, see the product documentation available with your HTTP Server.

Linux FTP Server

- Verify that the FTP server is running on the boot server.
- Verify that the FTP Server is enabled with anonymous access.

Configure FTP or HTTP Share for Linux Boot Server

If the boot server is on Linux, configure the FTP or HTTP settings before you activate the OS installation.

Note: Windows boot servers do not require any specific configuration.

Follow these steps:

1. Open DSM Explorer and navigate to Control Panel, Configuration, Configuration Policy. Select and unseal the policy that is applied on the boot server.
2. Navigate to DSM, Scalability Server, OSIM, ManagedPC, Server and specify the value for the following parameters depending on whether you want to configure HTTP, FTP, or both:

   Linux FTP anonymous root value

   Specifies the path where OSIM images are linked or mounted when shared through FTP URL.

   Default: /var/ftp

   Linux HTTP document root value

   Specifies the base path where OSIM images are linked or mounted when shared through HTTP URL.

   Default: /var/www
3. Save and seal the policy. Apply the policy on the boot server.

   The boot server is configured to use the base path you specified for HTTP or FTP.

Configure the Repository Servers

The repository servers contain native Debian software packages and OS installation files. Configure the repository servers (master and mirror) for HTTP or FTP access before deploying software or OS using Client Automation.

Note: You can use any FTP or HTTP server that is supported by your operating system.

Configure FTP Settings

Configure the FTP settings on the repository servers to help ensure that the FTP share is accessible for OS installations and software deployments.

Follow these steps:

1. Verify that the FTP server, vsftpd for example, is installed on the repository server.
2. Open the configuration file, /etc/vsftpd.conf for vsftpd, and verify that the following entries have the given values:

   `anonymous_enable=yes`

   `anon_root=`*`FTP_ROOT`*
3. Execute the following command:

   `mkdir` *`FTP_ROOT`*`/ubuntu`
4. Append the following line to the conf file /etc/fstab and save the file:

   `Mirror_location FTP_ROOT``/ubuntu none ro,user,bind 0 0`

   Note: Mirror_location is the path that you specified for the mirror in the mirror synchronization table.

   Example:

   `/var/spool/debmirror /var/ftp/ubuntu none ro,user,bind 0 0`
5. Execute the following command to export the repository to the FTP share:

   `mount -a`

   The command exports the repository at the /var/spool/debmirror directory to the /var/ftp/ubuntu share.
6. Log in as an anonymous user and access the FTP share.

   If the access is successful, it implies that the FTP settings are configured correctly.

Configure HTTP Settings

Configure the HTTP settings on the repository servers to help ensure that the HTTP share is accessible for OS installations and software deployments.

Follow these steps:

1. Log in to the repository server. Export the repository to the HTTP share using the following command:

   `ln –s` *`Mirror_Location HTTP_DOC_ROOT`*`/ubuntu`

   Note: Mirror_location is the path that you specified for the mirror in the mirror synchronization table.

   Example:

   ```
   ln –s /var/spool/debmirror /var/www/ubuntu
   ```

   The command exports the repository at the /var/spool/debmirror directory to the /var/www/ubuntu share.

2. Access the share from a browser on the local or remote computer using the following URL:

   `http://`*`hostname`*`/ubuntu`

   If the access is successful, it implies that the HTTP settings are configured correctly.

Chapter 11: LINUX OSIM OS Images

This section contains the following topics:

RedHat, SUSE, Oracle Enterprize Linux OS Images

This section describes special requirements for RedHat and SuSE OS images:

- The earlier Linux installations which the ITCM OSIM supports work with DOS osinstal.2,3 boot images.
- The new Linux installations which the ITCM OSIM supports work only with the Linux-based (customized CentOS 6.3 x64) boot image.
- For Linux distributions not including a loadlin.exe you have to acquire it from the internet and install it.

  Note: There are many places in the internet where you can download loadlin.exe and also many sites where the functionality and the installation of loadlin is documented. For example, see article "Loadlin" in the Wikipedia.
- Linux installations load the setup files from the Boot Server using a NFS share or TFTP.

- The Linux packages will be read from an NFS share.

  The NFS share is on the boot server but can alternatively be on a central NFS server.

  Depending on the preferred solution, the boot server or the central NFS server needs NFS running. On Windows, the “NFS Services” can provide the NFS server functionality.

- “NFS Services” must be installed on the server before creating or distributing the images.

- If the customer wants to have the Linux files and the NFS share on boot servers instead of the central NFS server, CreateOSImage or the wizard must be used without the –a option to create the LINUX image.

  CreateOSImage then inserts all files from the Linux CDs into the new OS-image. It also creates an NFS share with the sufficient access rights.

  RegisterOSImage then builds a Software Delivery package with the complete OS-image files. When distributing this Software Delivery package to a boot server, the installation procedure will copy all Linux files to the boot server and the installation will create a new NFS share.

- It is also possible to use an external NFS server (which is not the Boot Server).

For RHEL6 OS Image Creation:

When you use the Local or Network folder option in the OS image creation, ensure that the accessed folder has the following steps:

- RHEL6 ISO image is copied in to the folder and renamed to cd1.iso
- Copy the images folder from the RHEL6 ISO in to this folder
- Create a directory named Packages and copy the file Red_Hat_Enterprise_Linux-Release_Notes-6-en-US-3-7.el6.noarch.rpm from the Packages folder in the ISO.

  The version of the RPM package changes with RHEL6 updates.

- Copy the .discinfo file from the ISO image in this folder

  The directory structure is in the following format:

  SharedFolder\cd1.iso

  SharedFolder \images

  SharedFolder \ Packages\
  Red_Hat_Enterprise_Linux-Release_Notes-6-en-US-3-7.el6.noarch.rpm

  SharedFolder \.discinfo

Installing Linux from an External NFS Server

An external NFS server in the network providing an NFS share must contain the merged CDs or DVDs of the Linux distribution. Preparation of such a share is not part of OSIM.

For different LINUX distributions, the merged directory can look different. If you are not sure how that works, let createOSImge or the wizard create a complete image and copy the contents to the external NFS server. Consult the documentation of your LINUX distribution.

For example, OS which is delivered on five CDs, the files and directories from the five CDs are merged into one directory that becomes then the NFS share.

The files needs to be merged into the following structure:

```
susenfs

(This directory becomes the NFS share)
    -Boot
    -Dosutils
    -Media.1
    -Media.2
    -Media.3
    -Media.4
    -Media.5
    -Suse
        -I586
        -I686
        -Noach
        -Setup
```

Note: The susenfs share and the directories must get read access rights for ANONYMOUS USER.

To tell CreateOSImage that the created OS-image should use an external NFS share and not to copy all files from the CDs into the OS-image, call this command:

```
CreateOSImage -i<imagename> -o<OS-type> -a <NFS-share-name>.
```

But even using the –a option, CreateOSImage needs CD1 or "–s <path>" to copy some installation programs from the Linux distribution into the OSIM OS-image.

The given value "–a <image>" is set into the OSImageNFS boot parameter. The parameter OSImageNFS, as well as the address of the NFS Server, BootServerNFS, can be changed later in the DSM Explorer.

BootServerNFS must be set with the address or name of the NFS server providing the OSImageNFS.

Installing SUSE OS Images from an External NFS Server

The installation of all SUSE OS images is supported using only the extracted folders of ISO image for External NFS Server.

Installing RHEL6 OS Images from an External NFS Server

For deploying Red Hat Enterprise Linux 6 by using Centralized NFS Server, create the following folders and files under NFS Share folder:

- RHEL6 ISO image is copied in to the NFS share folder and renamed to cd1.iso
- Create a folder named images and copy the files install.img and product.img from RHEL6 ISO in this folder. The directory structure is in the following format:

  NFSShare\cd1.iso

  NFSShare\images\ install.img

  NFSShare\images\ product.img

Installing Oracle Enterprize Linux Images from an External NFS Server

For deploying Oracle Enterprise Linux (OEL) by using Centralized NFS Server, the following folders and files list are required to be created under NFS Share folder.

Follow these steps:

1. Copy OEL OS DVD ISO image to the Centralize NFS share folder and rename it to "cd1.iso".
2. Extract the folder "image" from OEL OS DVD ISO image and copy it to the Centralize NFS share folder.

Auto Answer Files for Customized OS Images

OSIM installs RedHat and SUSE OS images by using the Kickstart or AutoYaST installation methods. The auto answer files, provided by OSIM, are available as templates for frequently used installations. You can use OS installation parameters to customize the auto answer files to suit your requirements.

OSIM also has post procedures to configure the target installations and to install the DSM agent. You can use the OS installation parameters in the post procedures as well. This makes it possible to use only one OSIM image for a variety of installations.

More information:

responsefile= (see page 86)

Requirements for RedHat, SUSE, and Oracle Enterprise Linux OS Images

The requirements for RedHat, SuSE, and Oracle Enterprise Linux OS Images are as follows:

- An OSIM Image Prepare System (IPS)
- A Boot Server with an embedded NFS server or an additional external NFS server
- CDs or DVDs of the operating system (RedHat Enterprise Linux or SUSE Linux Enterprise Server)
- OSIM boot images such as osinstal.2, osinstal.3, or Linux-based boot image (CentOS 6.3 x64) based on OS Installation version.

Note: The OSIM IPS does not create new LINUX kernels to include special LINUX drivers. If the installation of the LINUX distribution requires a modified LINUX boot kernel, you must build the kernel outside OSIM, on a LINUX system and then add it to the OSIM image.

Methods for Creating and Installing the OSIM OS Images

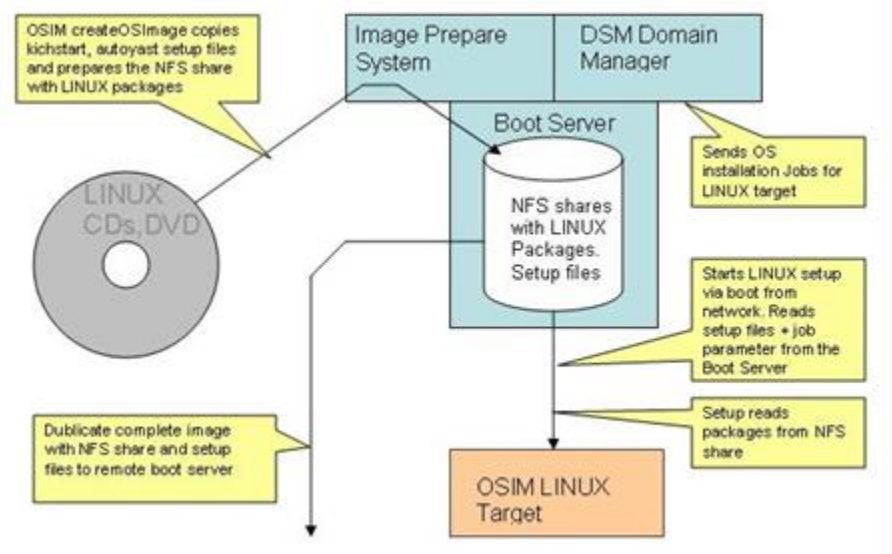
You can create and install OSIM OS Images for RedHat, SUSE, and Oracle Enterprise Linux depending on whether you want to keep all the files in the boot server or use an external NFS server to store the packages.

How to Create and Install the OS Images with all Files on the Boot Server

The following steps describe the process for creating and installing the OSIM OS image with all the files on the boot server. If you have an external NFS server, see How to Create and Install an OS Image with Package on an External NFS Server (see page 187).

1. Create the complete Linux OSIM OS image on the Image Prepare System, including all setup files, the boot kernel, and all packages.
2. The Image Prepare System creates a NFS share on the boot server, sets the appropriate access rights in the image store, and updates the OSIM job parameter OSImageNFS=<name of NFS share> with the new share name.
3. Register the Linux OSIM OS image as a software delivery package if you have additional remote boot servers. The LINUX OSIM OS image can be delivered through a software delivery job to additional remote boot servers.
4. The software delivery job prepares the NFS share that includes all packages and setup files on the remote boot server.
5. The boot server sets the job parameter BootServerNFS=<name or IP address of NFS server> to its own name when a target executes the OS install job.

The following diagram depicts this flow:

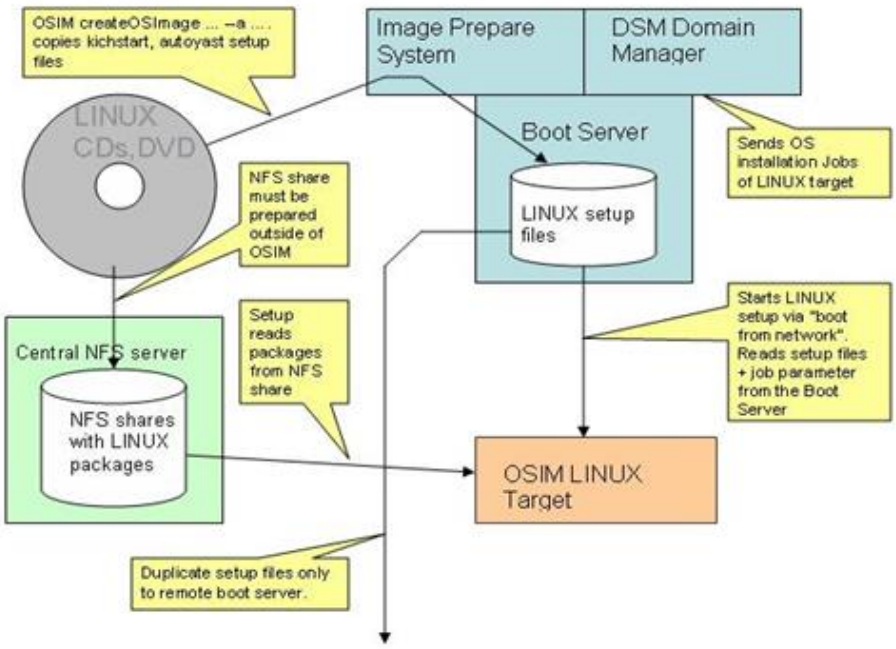


How to Create and Install an OS Image with Package on an External NFS Server

The following steps describe the process for creating and installing the OSIM OS image with the package on an external NFS server:

1. Create the Linux OSIM OS image on the Image Prepare System that includes only the setup files and the control files. This image can install the package from an external NFS server, which is not managed by OSIM.
2. Create a share on the NFS server and grant appropriate access rights.
3. Specify the name of the NFS server in the OSIM job parameter BootServerNFS=<name or IP address of the central NFS server>. The name of the NFS share will be taken from the createOSImage command and preset in the job parameter OSImageNFS=<name of NFS share>

The following diagram depicts this flow:



Create OSIM OS Images with all Files on the Boot Server

An OS image includes the operating system files required for the unattended installation of the OS at the target computers. Creating the OS image is the first step in automating the unattended installation of operating systems.

Depending on whether you want to create the OS Image from the CDs/DVD, or from a directory structure, execute one of the following commands:

CDs or DVDs

- **For RedHat Enterprise Linux**

  ```
  CreateOSImage -I <name> -o <REDHATES50(x64)-CD or REDHATES50(x64)-DVD
  ```

- **For SuSe Linux Enterprise Server**

  ```
  CreateOSImage -I <name> -o <SUSE102(x64)-CD or SUSE102(x64)-DVD>
  ```

The command reads the setup files from CD1 or DVD1 and copies the complete CDs or DVD as ISO file to the image store. If required, CreateOSImage asks for further CDs. In case of DVDs, insert the first DVD.

Directory Structure

- **For RedHat Enterprise Linux**

  ```
  CreateOSImage -I <name> -o <REDHATES50(x64)-CD or REDHATES50(x64)-DVD> -s <
  distribution files path>
  ```

  Distribution files path

  Specifies the path to the directory that includes at least all .iso files of the CDs or DVD, the directory images\pxeboot*, and the rpm package Server\syslinux*.rpm from CD1 or DVD1. For example, the directory should contain the following files:

  - \images\pxeboot*
  - \server\syslinux*.rpm
  - \.discinfo
  - \cd1.iso
  - \cd2.iso
  - \cd3.iso
  - \cd4.iso
  - \cd5.iso

- **For SuSe Linux Enterprise Server**

```
CreateOSImage -I <name> -o <SUSE102(x64)-CD or SUSE102(x64)-DVD> -s <
distribution files path>
```

Distribution files path

Specifies the path to the directory that includes at least all .iso files of the CDs or DVD, the directory boot*, and the rpm package suse\i586\syslinux*.rpm (for 32-bit) or suse\x86_64\syslinux*.rpm (for 64-bit), from CD1 or DVD1. For example, the directory should contain the following files:

- \boot*
- \suse\i586\syslinux*.rpm or suse\x86_64\syslinux*.rpm
- \cd1.iso
- \cd2.iso
- \cd3.iso
- \cd4.iso

Create OSIM OS Images which Installs Packages From an External NFS Server

1. Verify that the Linux distribution is available in ISO format. Rename the original ISO image file names in the distribution to cd1.iso, cd2.iso, ..., cd<n>.iso.
2. Create a share on the external NFS server and copy the ISO images from the OS CDs or DVD. The NFS share must contain at least all .iso files of the distribution, from cd1.iso to cd<n>.iso.
3. For the user "anonymous logon", grant read access to the shared directory and the .iso files in that directory on the NFS share. For more information, see the section Set Access Rights to External NFS READHATES5x and SUSE102 Shares.
4. Depending on whether you want to create the OS Image from the CDs/DVD, or from a directory structure, execute one of the following commands:

CDs or DVDs

- **For RedHat Enterprise Linux**

  ```
  CreateOSImage -I <name> -o <REDHATES50(x64)-CD or REDHATES50(x64)-DVD> -a <NFS share name>
  ```

- **For SuSe Linux Enterprise Server**

  ```
  CreateOSImage -I <name> -o <SUSE102(x64)-CD or SUSE102(x64)-DVD -a <NFS share name>
  ```

  NFS share name

  Specifies the name for the external NFS share.

  The command reads the setup files from CD1 or DVD1 and stores the files on the Image Prepare System / Boot Server.

Directory Structure

- **For RedHat Enterprise Linux**

  ```
  CreateOSImage -I <name> -o <REDHATES50-CD or REDHATES50-DVD> -s < distribution files path> -a <NFS share name>
  ```

  Distribution files path

  Specifies the path to the directory that includes at least the directory images\pxeboot\.discinfo and the rpm package Server\syslinux*.rpm from CD1 or DVD1.

- **For SuSe Linux Enterprise Server**

  ```
  CreateOSImage -I <name> -o <SUSE102(x64)-CD or SUSE102(x64)-DVD> -s < distribution files path> -a <NFS share name>
  ```

  Distribution files path

Specifies the path to the directory that includes at least the setup files boot* and the rpm package suse\i586\syslinux*.rpm (for 32-bit) or suse\x86_64\syslinux*.rpm (for 64-bit) from CD1 or DVD1. For more information about the folder structure, see Merging the Files from SUSE CDs or DVDs (see page 192).

NFS share Name

Specifies the name for the external NFS share.

Note: If you do not provide the syslinux*.rpm package, you must add syslinux.com later. For more information, see the section Adding Syslinux.com to the OSIM image.

Merging the Files from SUSE CDs or DVDs

Before creating the SUSE OS images, merge the files into a specific structure. The structure varies depending on the version of SUSE and the architecture.

Note: After merging the files, copy the parent SUSE folder to the NFS share.

SUSE 10 32-bit OS Images

suse

- boot
- docu
- media.1
- media.2
- media.3
- media.4
- media.5
- patches
- dosutils
- suse
  - il586
  - i686
  - noach
  - setup
  - x86_64

SUSE 11 32-bit OS images

suse

- boot
- docu
- media.1
- media.2
- suse
  - i585
  - i686
  - nosrc

- noarch
- setup
- src

SUSE 11 64-bit OS Images

suse

- boot
- docu
- media.1
- media.2
- suse
  - i586
  - noarch
  - setup
  - x86_64
  - nosrc

Creation of SUSE OS Images

The NFS share must contain the merged CDs or DVDs of the Linux distribution for SUSE versions 10 and 11, which is delivered on several CDs or DVDs.

Follow these steps:

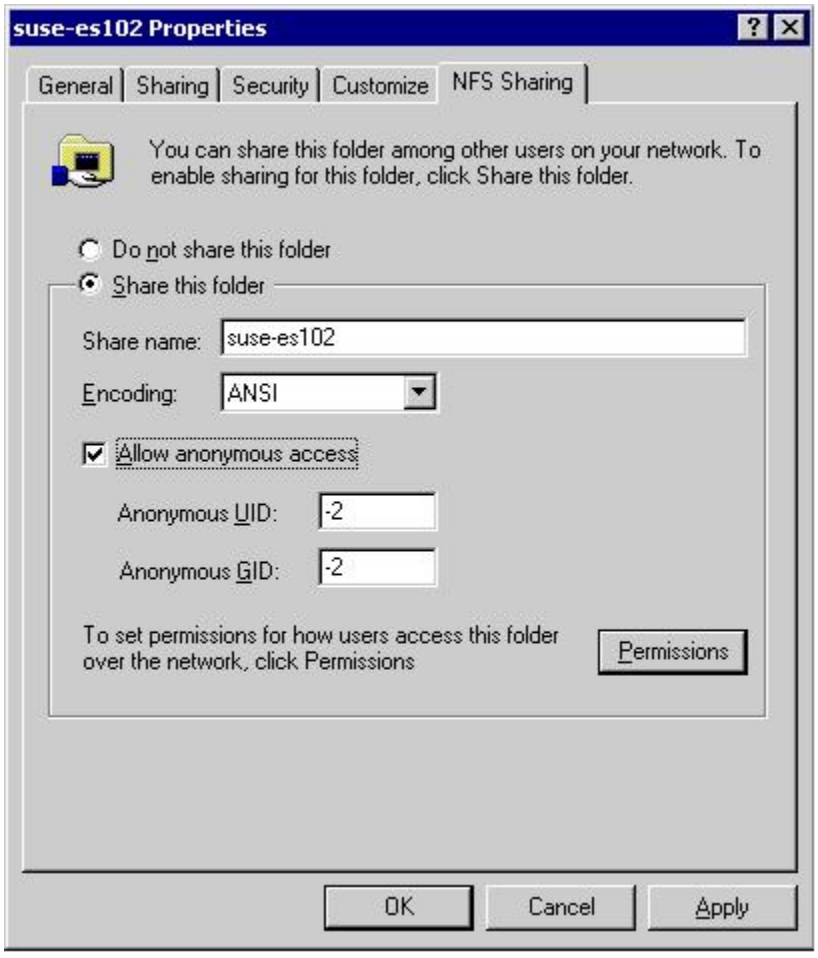
1. Merge the files and directories from the CDs or DVDs into one directory called suse. When merging the CDs or DVDs, start from the last CD or DVD so that CD1 or DVD1 is the last to be merged. For more information about the folder structure, see Merging the Files from SUSE CDs or DVDs (see page 192).
2. Copy the suse directory to the NFS share.

Set Access Rights to External NFS READHATES5x and SUSE102 Shares

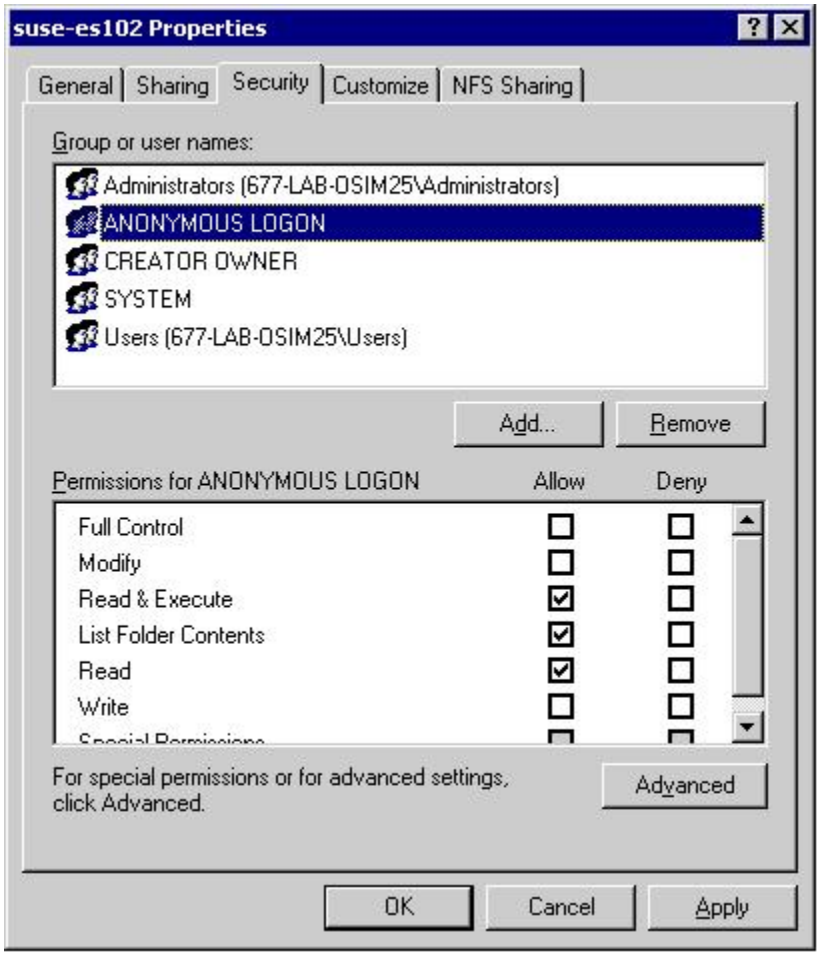
You must set specific access rights to the NFS share created on the external NFS server so that the target computer can access the image stored in the NFS share. Perform the following procedures based on whether your NFS server is on Windows or Linux.

To set the access rights on Windows NFS server running Microsoft UNIX services for Windows

1. Right-click the folder that you want to share on the NFS server, and click Properties.
2. Set the access rights as shown in the following illustration:



3. Click the Security tab and set the NTFS file system access rights as shown in the following illustration:



To set the access rights on Linux NFS server

1. Ensure that the NFS service is running using the following command:

   `Use /usr/sbin/showmount -e 127.0.0.1`

   This command checks whether the NFS server is active.

2. Configure the NFS share in /etc/exports. The entry looks like this:

   `/home/osim/redhat50-cd *(ro,all_squash,async).`

Add Syslinux.com to the OSIM image

When you run the createOSImage command to create an OSIM image, it searches the package for the file "syslinux.com". If the file is available, createOSimage command extracts the file into the new created image. If syslinux*.rpm does not include syslinux.com or syslinux*.rpm does not exist, createOSImage writes a file syslinux_readme.txt, which includes an error message. In this case, you must manually add Syslinux.com after the OSIM OS image is created.

Note: Some SUSE 10.2 and RedHat Enterprise Server 5x distributions include older syslinux.com versions that do not work on newer hardware. If you have problems with a hanging syslinux.com, download the latest version from the Internet. Tests on syslinux.com version 3.60 and newer were successful.

To add syslinux.com to the OSIM image:

1. Download the latest syslinux.com from http://syslinux.zytor.com/wiki/index.php.
2. Run the following command to show the path to the image directory:

   ```
   CreateOSImage -x
   ```

3. Copy syslinux.com to the image directory.

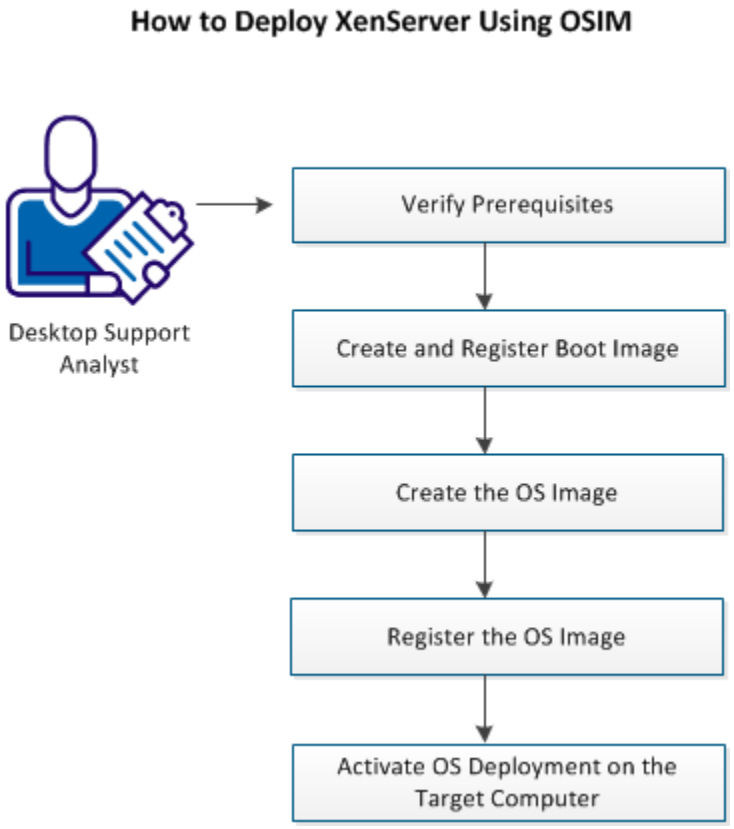
OS Installation Verification, Logging, and Troubleshooting

For information about verifying the OS deployment, see the Verify the OS Deployment (see page 158). For information about OS installation logging and troubleshooting, see the Logging and Troubleshooting (see page 159).

Chapter 12: How to Deploy Citrix XenServer Using OSIM

Client Automation helps you to implement unattended installation of Citrix XenServer operating systems on target computers using the OSIM functionality. As a Desktop Support Analyst, you deploy Citrix XenServer on new computers or those computers that have stopped functioning due to critical or irreversible failure.

The following diagram illustrates the steps that you perform to deploy Citrix XenServer using OSIM:



Perform the following tasks to deploy Citrix XenServer using OSIM:

1. Verify Prerequisites (see page 198).
2. Create and Register a Boot Image (see page 151)
3. Create an OS Image (see page 198).
4. Register the OS Image.
5. Activate OS Deployment on the Target Computer (see page 202).

Supported Citrix XenServer Versions and ISO Images

For the list of supported Citrix XenServer versions, see the Compatibility Matrix.

Verify Prerequisites

Ensure that you have working knowledge of Citrix XenServer Installation and Client Automation OSIM functionality.

Create and Register a Boot Image

You can create a Boot image by using the corresponding wizards in DSM Explorer or the corresponding commands from the Command Line Interface (CLI).

For more information, see Create and Register a Boot Image using DSM Explorer (see page 151) and Create and Register a Boot Image using CLI (see page 151).

Create an OS Image

An OS image includes the operating system installation files that are required for the unattended installation of the OS at the target computers. Create an OS image for the OS that you want to deploy using OSIM. You can create OS images using one of the following methods:

- Create an OS Image Using the Wizard (see page 199)
- Create an OS Image Using Command Line (see page 200)

For the OS images, the install source files can be configured for sharing from a central NFS server or the boot server. Verify that the appropriate permissions are set to allow unauthenticated read-only access.

Create an OS Image Using the Wizard

You can create an operating system image using the New OS Image Wizard. OS image creation from the wizard is straightforward though not as flexible as the command line.

Follow these steps:

1. Open DSM Explorer and navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, New OS Image.

   The Introduction page appears.

2. Click Next.

   The Operating System page appears.

   Select the image types depending upon the version of XenServer you want to use.

3. Click Next.

   The Image Name page appears.

4. Specify a name for the OS image and click Next.

   The Source page appears.

5. Specify the location of XenServer installation files by doing one of the following tasks:

   - Select the Install Linux from a central NFS Server option, if the XenServer installation files are stored centrally on an NFS Server. Specify the name of the share after selecting the option.
   - Clear the Install Linux from a central NFS Server option, when the XenServer installation files are stored locally on the NFS server that coexists with the boot server. Specify the files location from CD/DVD Drive or Local folder.

   Note: Verify that the Citrix XenServer CD is inserted and readable in a local optical device CD/DVD. The device can be a physical media or an ISO 9660 image that is mounted through a virtual device.

   Click Next.

   The Code Page appears.

6. Follow the instructions in the wizard and click Finish.

   The XenServer installation files are copied into the IPS and the boot server store is initiated. When the files are copied, the status is displayed in the last page of the wizard.

Create an OS Image Using Command Line

The OS image creation through the command line is more flexible than the OS image creation through the wizard. For example, you can specify an alternative location for the installation files.

To create a Citrix XenServer OS image using command line, execute the following command:

```
CreateOSImage {-i<imagename> -o<OStype> | -x | -d<imagename> | -p<imagename> |
-z<imagename> | -i<imagename> -e} [-s <OSpath>] [-t <driverpath>] [-l <locale>] [-k
<Prod.Key>] [-r <resp. file>] [-a <image>] [-g <imgtools>] [-n <path to language
packs>] [-h <path to hotfixes>]
```

Note: For more information about the parameters, see the OS Installation Management Administration Guide.

Example: Command for Citrix XenServer OS Images

The following examples create new Citrix XenServer 6.0 and 6.1 OS images that are named xens60 and xens61 respectively. The commands read the installation files from specified folders, and copy all installation files to the IPS target (does not use a central NFS share):

```
createosimage -i xens60 -o XENS-6.0-CD -s f:\xens-60-files
createosimage -i xens61 -o XENS-6.X-CD -s f:\xens-61-files
```

Register the OS Image

After the image is created locally on the IPS, register the OS image with the DSM domain to make it available for deployment.

Follow these steps:

1. Do *one* of the following steps to register the OS image:

   - Use the wizard:

     a. Open the DSM Explorer and navigate to Software, Boot and OS Image Library, Image Prepare System, Wizards, Register OS Image.

     b. Follow the instructions in the wizard and register the OS image with the domain.

     Note: When registering, you can opt to create an SD deployment image for distribution to one or more boot servers using Software Delivery.

   - Execute the following command:

     ```
     RegisterOSImage -s <manager> -i <imagename> | -w <directory> [-b] [-l] [-t] [-e] [-n <name>] [-v <version>] [-c <comment>] [-u <user> -p <password> -d <domain>]
     ```

     Note: RegisterOSImage can be started more than once on the same IPS, but it does not allow the registration of the same OS Image to different domain managers concurrently.

     Example: Registers and Selects an OS Image

     The following example registers an OS image from the Image Prepare System with the addressed OS Installation Manager and its associated domain manager:

     ```
     registerOSImage -s managername -i imagename
     ```

     Note: For more information about the command, parameters, and examples, see the OS Installation Management Administration Guide.

After the successful execution of the command, the OS image is registered with the domain manager.

Activate OS Deployment on the Target Computer

To perform an unattended installation of the OS, activate the OS deployment on the target computer.

Follow these steps:

1. Open DSM Explorer and navigate to Computers and Users, All Computers and find the target computer on which you want to deploy the OS.
2. Right-click the computer and select OS Installation.

   The OS Installation wizard opens.
3. Follow the instructions in the wizard and click Finish on the last page.

   An OS deployment job is created and the deployment status is displayed in the UI. You can access the UI from Computers and Users, All Computers, Group Details, and OS Installations.

OS Installation Verification, Logging, and Troubleshooting

For information about verifying the OS deployment, see the Verify the OS Deployment (see page 158). For information about OS installation logging and troubleshooting, see the Logging and Troubleshooting (see page 159).

Chapter 13: OSIM OS Images for VMWare

Creating an OS Image

You can use two different methods to create an OS image for VMware ESX on the Image Prepare System and installed from a boot server.

- Copy the VMware ESXI DVD(CD) manually into an NFS share.
  The created OS image for VMware ESXi contains only files that are required to start the Linux boot kernel and to control the OS installation process.
- Use the VMware ESXi software packages that are embedded inside the OS image.
  The OS image contains the entire DVD(CD) contents.

You can create an OS image on the Image Prepare System either with a command-line utility from a command prompt window or by using the DSM Explorer.

Create OS Image Command:

This command requires the type (-o option) and the name (-i option) of the OS image to be created. The new introduced types for VMware ESXi are:

```
createosimage –o VMWARE-ESXi51 –i esxi51
```

Boot Server

A boot server hosting an OS image of VMware ESX must have activated the NFS service. On a Windows boot server, install the Microsoft Windows Services for Unix and start the Server for NFS.

Steps and Considerations

- Use complex passwords when setting a password for ESX. See VMware password requirements and restrictions.
  If you do not use a complex password, then the root password is not correctly set to the desired value and the active value is the string default. The OSIM install log records the failure to set a password, but does not fail the job as a whole.
- Installation on Cisco UCS Systems and other systems with multiple disks. The default OSIM ESX installation deletes all partitions from the first detected hard disk only. If the target hardware contains two or more disks and the ESX installation detects that any VMFS partitions are present, then the installation aborts. This behavior by the installer ensures that any virtual machines are not destroyed.
  If it is required to provision ESX servers that already have existing VMFS partitions on non-primary disks, then modify the kickstart file such as ks.cfg, to add the overwitevmfs switch to the clearpart command, as follows:
  `clearpart --overwritevmfs –-firstdisk`
- The ESXi host hypervisor environment does not support installation and/or operation of the native DSM agents. Use the AM RVI module for discovery of ESXi instances. Installation of a native DSM agent is supported on a guest running as ESXI virtual machines, dependent on the installed operating system and version.

OS Installation Verification, Logging, and Troubleshooting

For information about verifying the OS deployment, see the Verify the OS Deployment (see page 158). For information about OS installation logging and troubleshooting, see the Logging and Troubleshooting (see page 159).

Chapter 14: Unsigned Drivers Support for Windows

CA Client Automation release 12.9 supports the installation of unsigned drivers to Windows PE boot images and ImageX Windows OS images. This support is available on the following platforms:

- Windows 7
- Windows 7 SP1
- Windows 2008 R2
- Windows 8.0
- Windows 2012
- Windows 8.1
- Windows 2012 R2

With the earlier releases, creation of a 64 bit WinPE image with unsinged drivers would fail and the ImageX deployments would either prompt the user for acceptance or result in a failure, based on the OS settings related to driver signing requirements.

Boot Image Support

The boot image templates of WinPE30x64 and WinPE40x64 are extended to support unsigned drivers while creating the corresponding boot images. If you are upgrading, you can include any unsigned drivers in the existing boot images using Update Boot Image wizard from the IPS UI. You can also include the unsigned drivers using command line interface.

Note: The 32 bit boot images do not need any additional changes for supporting unsigned drivers.

OS ImageX Support

The GETIMAGE-WIN8 and GETIMAGEX64 operating system types are extended to support the unsigned drivers for ImageX OS deployments. These types use Windows ADK 8.0 for creating OS images. To include unsigned drivers in the corresponding ImageX OS images, the ADK version associated with those operating systems must be ADK 8.0 or earlier.

For example, the ImageX images that are created from WIM files using GETIMAGEX64, can only include unsigned drivers for ImageX deployment of the following operating systems:

- Windows 7
- Windows 7 SP1
- Windows 2008 x64
- Windows 2008 R2
- Windows 8
- Windows 8 .1
- Windows Server 2012
- Windows Server 2012 R2

You can use the same getimagex64 OS to deploy ImageX of Windows 8.1 and 2012 R2, by copying all the DISM binaries from the Windows ADK 8.1 install location and replace the existing ones in the "imagex" folder of the created getimagex64 OS. That is, copy the binaries from %programfiles%\Windows Kits\8.1\Assessment and Deployment Kit\Deployment Tools\amd64\DISM to

```
<DSMInstallDir>\DSM\Server\SDBS\var\managedpc\images\<OSImageName>\<OSImageName>\imagex folder.
```

Alternately, you can also copy the DISM binaries to the OS image folder of the Windows 8.1 and 2012 R2 IMAGEX images.

The default path for IMAGEX images is:

```
<DSMInstallDir>\Server\SDBS\var\managedpc\images\<OSImageName>\<OSImageName>.
```

If you are using GETIMAGE-WIN8, then copy the DISM binaries from the X86 folder. Typically following is the default path:

```
%programfiles%\Windows Kits\8.1\Assessment and Deployment Kit\Deployment Tools\x86\DISM
```

Notes:

1. In case of upgrade, the existing images of GETIMAGE-WIN8 , GETIMAGX64 and their corresponding ImageX types cannot be updated using IPS tools to support the unsigned drivers installation.

2. If unsigned driver support for IMAGEX deployments is needed only till Windows 8 and Windows 2012, then copy the following DISM binaries from ADK8 installation folder to %programfiles%\Windows Kits\8.1\Assessment and Deployment Kit\Deployment Tools\<OSArch>\DISM.
   - imagingprovider.dll
   - folderprovider.dll
   - vhdprovider.dll
   - compatprovider.dll
   - ssshim.dll
3. If we intend to support windows 8.1 and 2012 R2 as well, then DISM binaries from ADK 8.1 need to be copied as mentioned earlier.

Chapter 15: Troubleshooting

This section contains the following topics:

Log File Collection Tool dsminfo

CA Technologies provides the dsmInfo tool, which collects diagnostic information from systems that have Client Automation installed. The data collected is compressed into a single file that contains log files, system information, directory structures, and registry and environment information. This diagnostic tool is available in the Client Automation product installation media under the DiagnosticTools folder.

If a problem with Client Automation is reproducible, then run the following command to change the trace level to DETAIL:

```
cftrace -c set -l DETAIL
```

Reproduce the problem and collect the diagnostic information with the dsmInfo tool.

Notes:

For more information about this tool, see the DSMInfoReadMe.txt file available under the DiagnosticTools folder in the product installation media.

The dsmInfo tool produces ".7z" files by default. These files provide better compression than zip files, so uploading to CA Technologies is easier.

Server with Multiple NIC Cards

Symptom:

The DSM Domain Manager is running on a server with multiple NIC cards, each configured for different subnets. Does one need a scalability server on those different subnets?

Solution:

An OSIM boot server always serves all subnets it is directly attached to by NICs. This also holds for the boot server on a domain manager. So in case of a multiple NIC domain manager additional scalability server are not necessary for those subnets addressed by the multiple NICs. Additional boot servers or a forwarding of DHCP requests is only needed for those subnets not directly attached to the domain manager.

Operating System Deployment Fails on VMware Virtual Machine Targets

Symptom:

When I use VMware virtual machine with SCSI Controller type VMware Paravirtual (PVSCSI) as a target for OS Deployment, OS deployment fails with the following error message:

```
No Local disk attached to the system
```

Solution:

Do one of the following based on your Operating System deployment:

For Citrix Xen Server:

- Select a type other than PVSCSI as a SCSI Controller.

For Red Hat Enterprise Linux, Oracle Enterprise Linux, SUSE Linux Enterprise, and Kubuntu:

- Select appropriate drive name as Install Drive parameter, for example, sda, sdb.

Wrong Sysprep Version Windows 7 with OSIM Getimage: Prepare Phase

Symptom:

When I create a GETIMAGE for Windows 7 and run the prepare phase. WinPE boots but fails with the following error:

Wrong sysprep version windows 7

Solution:

This behavior is due to a hidden partition on the HD prior to the C:\ which is often created by bitlocker or a vendor's build CD. The getimage runs a script to detect the first partition and then looks for the sysprep version to ensure it is correct. When there is a hidden partition the script looks here instead of on the c: and thus cannot find sysprep.

Build the template without the hidden partition. The simplest way to ensure this is to build the template via OSIM using a standard image. The OSIM process runs diskpart which will remove all the partitions and install the OS to a single partition. As we do not recommend imbedding the agent in the image you should select Installagent=no while building the template PC in this manner.

Boot Server Hostname Limitation

Symptom:

The deployment of Windows OS using Linux boot server failed during the boot image phase with the following error:

The network path was not found.

or

Fatal Error: No camenu-11 Boot Server share available.

Solution:

This behavior is due to the length of the boot server hostname. Deployment of Windows OS fails as the Windows client fails to access the file on the boot server hostname that has more than 15 characters. You can fix this issue by restricting the boot server hostname to 15 characters long only.

Alternative DOS Boot Loader bootdos.f12

Symptom:

The installation of a new operating system starts immediately. This behavior is not desired in some cases.

Solution:

With the alternative DOS boot loader bootdos.f12, the user in front of the target can press CTL F12 within a 4-second time frame to start the OS installation.

Otherwise, bootdos.f12 will boot from the local hard disk and the OS installation job is postponed to the reboot when pressing CTL F12.

To use the bootdos.f12 in a sequence of reboots during an OS installation, the first boot image osinstal.2 has to be changed into an indirect boot image with its own loader.

To change the boot image osinstal.2 into an indirect image

1. Create a directory osinstal.2 in the boot image store of the Scalability Server (Boot Server).
2. Create a subdirectory osinstal.2\undi.
3. Move the DOS boot image UNDI\osinstal.2 to osinstal.2\undi\osinstal.2.

4. Copy the boot loader bootdos.f12 from the boot image store into the osinstal.2 directory.
5. Edit a description file named osinstal.2 in the UNDI directory according to "OS Boot Image Extension for Boot Images Consisting of Multiple Files."

Contents of the osinstal.2 description file:

```
[BootImage]
Boottype=DOS
BootImageDir=osinstal.2
BootLoader=bootdos.f12
SwitchFile=osinstal.2
```

Images:

DOSBOOT //Boot image store

UNDI //RAM disk image files DOS, WINPE, LINUX

osinstal.2 //description file

ostinstal.3 //real 1.44 Mb DOS floppy image

osinstal.2

bootdos.f12

undi

/osinstal.2 //real 1.44 Mb DOS floppy image

Boot Server Linux Additional Requirements

Symptom:

When I use Samba shares there is the following error message “ERROR: Failed to initialize locking database“ in the ca-dsm.log file.

Solution:

If the boot server is installed on a Linux computer with custom setup, it is possible to enable share access. The Linux boot server then looks for SAMBA in order to create OSIM shares.

In this case, enable SAMBA on your Linux system before starting the boot server installation.

Samba must be configured to use share security and encrypted passwords.

With some older SAMBA versions, the SAMBA configuration program produces the ERROR message “ERROR: Failed to initialize locking database“ in the ca-dsm.log file.

Please ignore this message, because SAMBA normally is configured properly.

SAMBA Error with the Scalability Server (Boot Server) Installation on Linux

Note: SAMBA servers may display an error message with the scalability server (boot server) installation on Linux.

When a boot server is installed on a Linux computer with Custom Setup, it is possible to enable share access. If share access has been enabled, the Linux boot server looks for SAMBA to create OSIM shares.

With some older versions of SAMBA, the configuration program produces the ERROR message “ERROR: Failed to initialize locking database“ in the ca-dsm.log file. Ignore this message, because SAMBA is probably configured properly from an OSIM point of view and can be used.

OSIM Linux Boot Server and Samba Netbios Names

Symptom:

I have a problem with the generated SAMBA NETBIOS names.

Solution:

If a Linux boot server uses the share access method, the SAMBA NETBIOS-Name is generated from the computer name. In this case, the computer name may not exceed 15 characters.

Boot Server Share Access Restrictions on Windows Server 2008

Symptom:

Accessing the boot server's DOS share from the OSIM client fails with an "Access Denied" error message or results in "Error 93".

Solution:

You need to configure the security settings of the boot server share.

Follow these steps:

1. From the Control Panel, navigate to Administrative Tools, Local Security Policy, Security Settings, Local Policies, Security Options.

   The security settings pertaining to the local security policy appear.

2. Ensure that the following policies have the required settings as given below:

   - Microsoft Network server: Digitally sign communication (always) is disabled.
   - Network security: Do not store LAN Manager hash value on next password change is disabled.

     Note: You must restart the boot server after changing this configuration.
   - Network security: LAN Manager Authentication level is set to one of the following values:
     - Send LM & NTLM responses
     - Send LM & NTLM - use NTLMv2 session security if negotiated
     - Sent NTLM response only
     - Send NTLMv2 response only

Remote Boot Server Does Not Provide Software Delivery Agent Packages

Symptom:

The OSIM targets remain in the "installing" state though no DSM agents are installed. My boot server does not provide required Agents for automatic installation with OSIM.

Solution:

By default, a remote scalability server does not provide agents for automatic installation with OSIM. Use the DSM explorer for staging the Agent package from the All Software Library to the remote scalability or boot server.

For Windows OSIM targets stage the package "CA DSM Agent + Software Delivery plugin <version>.

For LINUX OSIM targets stage the package "CA DSM Agent + Software Delivery plugin Linux(intel) [ENU] <version>.

Credentials Needed for Registerosimage and Registerbtimages

Symptom:

Cannot launch the commands RegisterOSImage and RegisterBtImages.

Solution:

RegisterOSImage and RegisterBtImages need credentials to log into the domain manager. The default is your login user. If this user does not have the rights to register OS boot images, please use the -u <user> -p <password> -d <domain> parameters.

If you are not sure whether you are logged in with an authorized user, enter the same credentials as in the DSM Explorer for the connection to the manager.

OS Installation Fails While Downloading Installer Components

Symptom:

The OS installation on the target computer fails at the Download Installer Components screen with the following error:

```
No Kernel Modules were found. This is probably due to a mismatch between the kernel
used by this version of the installer and kernel version available in the archive.
```

If you are synchronizing a distribution with a name that ends in -updates, the debmirror utility skips the synchronization of the debian-installer component. This issue occurs due to a bug in the debmirror utility.

Solution:

To resolve this issue, edit the debmirror perl script to remove '.*-updates' from function 'di_skip_dist'. This workaround is applicable for debmirror 2.10 and 2.12 versions."

OS Installation Completes but Status Remains in Installing

Symptom:

Though OS installation on the target computer completes, the status of OS installation remains in Installing. This problem can occur due to a known network configuration issue with the Kubuntu OS. For more information about this issue, see https://bugs.launchpad.net/ubuntu/+source/network-manager/+bug/995165.

Solution:

Open the preseed.cfg file under *DSM Installation Directory*\Server\SDBS\var\managedpc\images*Image Name**Image Name*\IMAGESUP and add the following code at the end of the file:

```
$InstallAgentComment$ sed -i -e 's/iface eth[0-9] inet6 auto/# Replaced by OSIM # &/'
/target/etc/network/interfaces; \
```

The code comments out the detected IPv6 SLAAC addresses from interfaces so that NetworkManager can manage the interface, and the IPv4 and IPv6 stacks upon the reboot.

OSIM Installation of XenServer Fails

When the boot server runs on a Microsoft Windows server, depending on the OS version, it is possible that the NFS server requires configuration changes. Perform the following steps to ensure successful OSIM installation of XenServer:

Windows Server 2008 and 2008 R2

Modify the NFS services as described in the following CA Support Knowledge Base entry:

- http://support.microsoft.com/kb/977517/en-us

Files Added to OS Images Must Not Be Read-only

Symptom:

When I add MSI installers, drivers, and the like, to my OSIM OS image, the image will not be created.

Solution:

If you add files to OSIM OS images like MSI installers, drivers, service packs, and the like, you must make sure that these files are not read-only. Otherwise the image cannot be created.

Problem with Size Details of ITCM and CIC Components on the Add/Remove Programs

Symptom:

After I install or upgrade to Client Automation Release 12.9; Control Panel, Add/Remove Programs does not display the size estimate of Client Automation, Content Import Client (CIC), Patch Manager, and other Client Automation components.

Solution:

Obtaining and displaying the size in Control Panel, Add/Remove Programs is a property of Windows OS and not of InstallShield. This behavior occurs due to a change in the functionality with how Microsoft calculates the estimated size on Windows. Size of each installed component is estimated using the OS algorithm. This algorithm can vary in different versions of windows; impacting the display of Client Automation size estimates.

To populate the size estimates, you can perform the following steps:

Follow these steps:

Microsoft exclusively uses *EstimatedSize* registry key to populate the estimated size value in Add/Remove Programs. To populate the size estimates, you can manually edit *EstimatedSize* registry key in HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{PRODUCT_CODE}. See Microsoft documentation for more information.

Boot Sequence for OSIM Deployment

For OSIM, boot sequence is an important factor. If you deploy an OS into a system which contains multiple disks (either local or remote, or both local and remote), then, the disk being used for the OS deployment must be the first disk in the boot sequence among all the available bootable disks (disks holding any bootable OS or bootable media).

HostUUID Must Be Removed on ImageX and Ghost (16-bit and 32-bit) Images

Symptom:

If the image contains the DSM Agent **(not recommended!), the** registry of an ImageX or a Ghost image contains the DSM HostUUID-key. Multiple systems installed from this image will get the same HostUUID. This will cause management problems.

Solution:

Remove the HostUUID key from the registry for any ImageX or Ghost image containing a Client Automation object that includes the Client Automation agent or the Data Transport Service agent.

The key to be removed is found in the following location:

`HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\HostUUID.`

Createbtimages: Floppies and eTrust

Symptom:

The command CreateBTImages does not work in my eTrust-protected environment.

Solution:

When you run the command CreateBTImages in an eTrust-protected environment, it is necessary to grant the command full access to the floppy disk.

OS Installation Parameter OSUser Must Not Be "administrator" or "guest"

Symptom:

When I install an OS image with OSUser parameter "administrator" or "guest" the installation does not succeed on Windows 2000, Windows XP, and Windows Server 2003.

Solution:

The installation of Windows 2000 and Windows XP/Windows Server 2003 does not accept an OSUser parameter set to "administrator" or "guest".

The OSUser parameter is used to personalize the operating system software with Name and Organization and is restricted by Microsoft.

Problems with DOS LAN Manager Client in DOS Boot Images

Symptom:

Cannot install an OS image from a network share when it uses DOS boot images containing a MS client.

Solution:

The OSIM target uses a boot image including a MS Client to install the OS image from a network share. On some platforms you cannot run the MS Client or the setup of the OS image has not enough free RAM when the MS Client is loaded.

In such cases, we recommend to use WinPe or - if this is not possible (Linux) - to switch the responsible boot server to TFTP download method.

Then, the boot image will not load the MS client, but will use the small tftp.exe program for download.

DSM Explorer Can Change the Password Values in OS image Defaults

Symptom:

When I install OS images the default password is used which is not desired.

Solution:

In DSM Explorer, you can change the Password Values in OS image Defaults

If a domain password has been changed, it is possible to use the new password for all new OS installations.

All OS parameter values of the pwd (password) can be changed in the default parameter values in the DSM Explorer, even if the OS Image is assigned to computers.

If the default password is changed in an already assigned OS image, the new password will be used immediately in all affected configurations except the configurations where the default password is overwritten with a computer specific password.

Note: If the new default password was changed after a configuration became pending, the new password will not be downloaded to the boot server in order to become the new password for the pending OS installation.

OS Installation Cannot Detect NFS Server

Symptom:

OS installation does not detect the NFS server.

Solution:

If the OS installation can not detect the NFS server, the IP address has to be used for the parameter “BootServerNFS” .

The parameter “OSImageNFS” must be set with the NFS share name providing the Linux image on the NFS server.

If the LINUX setup ends with access denied, check the NFS share access rights. ANONYMOUS USER must have read access. Check also the file, directory access rights of the shared files, directories. ANONYMOUS USER must have read access.

Overflow in Sharing BufferWhen Using DOS Boot Images

Symptom:

Installation using share based access stops with the following error message on the target machine:

‘Error 36: The system has detected an overflow in the sharing buffer.’

Solution:

When installing OS images using share based access to the boot server, on certain types of target hardware the installation stops with this message. In this case use WinPE or switch the responsible boot server to TFTP access mode.

Ghost Image Restrictions

When creating a Ghost image for use with OSIM the following restrictions apply:

- The name of the Ghost image file must be a valid 8.3 DOS filename.
- If a FAT16 system partition is used to create the Ghost image, its size must not exceed 2 GB. Otherwise use FAT32 for the master partition.

Note: If the Ghost image is created from the system partition only - as opposed to creation from the whole disk - the "CreateNewPartitions = no boot" parameter will preserve existing partitions at the target machine other than the system partition.

During conversion of the new system partition from the temporary FAT16 to NTFS it is expanded to the previous size of the system partition. This means, if there are no other partitions on the target disk, the whole disk will be used.

With "CreateNewPartition = yes", the partition schema in <imagename>.par will be created on HD before the Ghost image is restored.

Do Not Use Unicode Multi-language Characters in the Workgroup Parameter

Symptom:

OS image deployment over Samba fails if boot parameters contain Japanese characters.

Solution:

Do not use Unicode Multilanguage characters in the Workgroup parameter. The DOS MSclient from Microsoft does not accept Unicode characters in the Workgroup parameter. If you use Unicode characters in the Workgroup boot parameter, the boot server shares cannot be opened from the DOS boot image running on the target computer.

TimeZone Parameter Truncates Values

Symptom:

When I am using an OS image, for example, a Vista image, I noticed that the OS parameter TimeZone is truncating values.

Solution:

This happens if the Timezone parameter contains blank values.

For a Vista image, change the setting of the TimeZone parameter to "\"W. Europe Standard Time\"" or "\"W. Europe Standard Time\" UK time"

Example: Modify OSIM install parameter TimeZone

This example shows how to change the parameter TimeZone for a Vista image.

```
cadsmcmd targetcomputer action=modifyInstallParameter
    name=computer name
    paramname=TimeZone paramvalue="\"W. Europe Standard Time\""
```

Note: If the parameter is a map list or extended map list, use the format "<key> <comment>". The key and the comment are separated by a blank, and the comment is optional. And if the key contains blanks, enclose it in quotes.

Boot Server or DHCP Server Not Functioning Properly

Symptom:

If you have the DHCP server and the boot server on the same system, one of the servers may stop functioning properly. This is because both the DHCP server and the boot server listen at UDP port 67.

Solution:

If you have installed the DHCP server and the boot server on the same machine, ensure to either disable the boot server at the time of installing the manager or scalability server, or configure the boot server to stop listening at port 67. You must also configure the DHCP server so that a booting PXE client can be notified that there is a listening boot server on the network.

To configure the boot server to stop listening at port 67

1. Create a configuration policy for the boot server, if one does not exist, and modify the following policy settings under DSM, Scalability Server, OSIM, ManagedPC, Server:
   - Set "Enable DHCP proxy" to "False"
   - Set "Use answer control list" to 0
2. Apply the policy to the boot server that co-exists with the DHCP server.

To configure the DHCP server

Configure the DHCP server by adding option 60 (class identifier) to the responses that the DHCP server sends to PXE clients. Depending on the operating system on which DHCP server is installed, perform the following steps:

Windows

1. Open DHCP Configuration and Management, right-click the server, and select Set Predefined Options.
2. Add option 60 to the defined options list with the value as "PXEClient".

Linux

1. Set the value of "vendor-class-identifier" option to "PXEClient"

As described in the Preboot Execution Environment (PXE) Specification Version 2.1 (1999 by Intel Corporation), the PXE clients will now contact the boot server on the DHCP server's system using UDP port 4011 to get boot instructions. This enables the OSIM Boot Server to serve the PXE clients. There is no dynamic OSIM boot server assignment in this case. The OSIM boot server will not serve PXE clients that have been served by a different DHCP server, unless forwarded to the Boot Server using option 43.

Note: With PXE 2.x clients, it is still possible to assign a different boot server via DHCP option 43 (encapsulated vendor-specific options) as described in Tech Document TEC381737.

DHCP Server Does Not Provide Boot Server Information

Symptom:

The DHCP server is not providing the boot server information.

Solution:

This happens if your boot server is not located on the DHCP server.

With PXE 2.x clients, you can assign a different boot server using the DHCP option 43 (encapsulated vendor-specific options). The DHCP server must be configured to provide additional boot server information in the DHCP offer using option 60 and option 43:

- Option 60: vendor-class-identifier
- Option 43: with the suboptions 6,8,9,10,71

After the PXE client has received an IP address from the DHCP server, the PXE client must send a DHCP Request to the specified boot server (in option 43:Suboption 8: PXE boot servers) to get the boot image file name.

Example: DHCP Offer

```
Option 53: DHCP Message Type = DHCP Offer
Option 54: Server Identifier = 192.168.111.131
Option 60: Vendor class identifier = "PXEClient"
Option 97: UUID/GUID-based Client Identifier (17 bytes)
Option 43: Vendor-Specific Information (PXEClient)
    Suboption 6: PXE discovery control = 2
    Suboption 8: PXE boot servers (7 bytes)
    Suboption 9: PXE boot menu (38 bytes)
    Suboption 10: PXE menu prompt (24 bytes)
    Suboption 71: PXE boot item (4 bytes)
    End PXEClient option
End Option
option 43 in detail:
2b 55 06 01 02 08 07 00 07 01 c0 a8 6f 83 09   +U.........o..
26 00 07 23 43 41 2d 55 6e 69 63 65 6e 74 65 72  &..#CA-Unicenter
20 4d 61 6e 61 67 65 64 50 43 20 42 6f 6f 74 20   ManagedPC Boot
53 65 72 76 65 72 00 0a 18 00 50 72 65 73 73 20  Server....Press
46 38 20 74 6f 20 76 69 65 77 20 6d 65 6e 75 00  F8 to view menu.
2e 47 04 00 08 00 00 ff                          .G......
```

In the above example, c0 a8 6f 83 is the IP address of the OSIM boot server.

Notes:

- The detail of option 43 above shows the complete hex string of the option.

- When defining the value for option 43, do not include the first 2 bytes (2b 55) as these are added by the DHCP server. (2b is the descriptor for option 43 and 55 is the length).
- Ensure that the DHCP offer should not contain option 67: boot file name
- Option 60 is not a standard option and must be manually added using netsh.exe

To add DHCP option 60 to DHCP server (Windows 2000 or later)

1. Open the command prompt.
2. Type netsh.
3. Type dhcp.
4. Type server *servername* or server *ip_address*

   A command prompt with the title dhcp server appears.
5. Type the following commands for PXE support:

   ```
   add optiondef 60 PXEClient STRING 0 comment=option added
   ```

   ```
   optionvalue 60 STRING PXEClient
   ```
6. Type the following command to confirm that everything has been set correctly:

   ```
   show optionvalue all
   ```

PXE-32: TFTP Open Time-out

Symptom:

OS installation fails immediately after completing the DHCP or PXE phase with an error message "PXE-32: TFTP open time-out" at the target computer. This problem occurs if the network did not support the packet size requested by the PXE firmware on the target or the MTU size is reduced due to tunneling. IP packets exceeding the MTU size of the network are transmitted in several fragments. Fragmentation is handled by the network layer and typically transparent to the application. Some PXE firmware implementations and also the TFTP client used in DOS boot images do not support fragmentation. Therefore, the effective packet size must be restricted by the boot server in those cases.

Windows PE based boot images can handle fragmentation and as a rule are not affected by a reduction of the MTU size.

Solution:

Restrict the effective TFTP blocksize used by the boot server using the configuration policy.

Note: From Release 12.5, the OSIM target program, sdmpcimg, used by Windows PE boot images requests specific TFTP packets, which are not affected by the configuration parameter TFTP blocksize limit. The parameter TFTP specific blocksize limit can be used to limit the effective blocksize for specific TFTP packets. For more information, see sdmpcimg Command (32bit and 64bit Version) (see page 141).

To restrict the TFTP blocksize

1. Open the configuration policy that is assigned to the boot server.
2. Navigate to DSM, Scalability Server, OSIM, ManagedPC, Server, and change the TFTP blocksize limit to a suitable value. For example, if the firmware requests 1456 TFTP data bytes, it may be sufficient to set the TFTP blocksize limit to 1350.

   Note: DOS boot images and Client Automation r11.x or r12 Windows PE boot images do not support a TFTP blocksize < 1024.

Deployment of Red Hat Enterprise Linux 5 Update 5 OS Fails

Symptom:

In OS Installation Management, when I deploy Red Hat Enterprise Linux 5 Update 5 with an Image created using CDs media, OS installation fails with the following error:
Red Hat Installation tree in that directory does not match your boot media.

Solution:

Do the following:

- While creating Red Hat Enterprise Linux 5 Update 5 OS Image in Image Prepare System, use DVD as a source media.
- While creating OS Image in Image Prepare System, use the following options:
  - REDHATES50-DVD Red Hat Enterprise Linux 5.x x32 (DVD + syslinux.com)
  - REDHATES50x64-DVD Red Hat Enterprise Linux 5.x x64 (DVD + syslinux.com)

OSIM Samba Shares not Enabled during Boot Server Installation

Symptom:

Sometimes OSIM Samba shares are not enabled on Linux Operating system even the option "Disable use of windows network shares" is deselected during the installation.

Solution:

Use the following commands to enable Samba shares after the boot server installation:

- Use the following command to switch to the tfpt mode:
  `sdbsswitch -t`
- Use the following command to switch back to the share method that creates the required changes in the Samba configuration file,:
  `sddbswitch -s`

Unattended Installation of SLES 11SP1 with DHCP

Symptom:

When I configure an unattended install of SLES 11SP1 to use DHCP address allocation without an explicitly stated domain name entry in the autoinst.xml file, then the installer discards any supplied host name and selects a randomly generated hostname, similar to linux-6dyj.

Solution:

To allow for a fixed hostname with the DHCP assigned address, modify the OS image-specific autoinst.xml file to include a default domain in the networking/dns section, as follows:

```
<networking>
  <dns>
    <dhcp_hostname config:type="boolean" >false</dhcp_hostname>
    <dhcp_resolv config:type="boolean" >true</dhcp_resolv>
    <hostname>$HostName$</hostname>
    <domain>forward.inc.local</domain>
  </dns>
</networking>
```

You can find the autoinst.xml file at the following location:
DSM_Install_Folder\server\SDBS\var\managedpc\images\IMAGE_NAME\IMAGE_NAME\suse

Change the Configuration to Suit the Microsoft NSF Server

OSIM requires that the boot server hosting XenServer installation files must be exported through NFS services. Based on the OS version, change the configuration to suit the NSF server when the boot server is running on the Microsoft Windows Server.

- Windows Server 2008 and 2008 R2

  For detailed instructions on how to modify the NSF services, see Microsoft documentation KB entry:977517

Windows Detects Error while Installing Boot Image

Valid on Windows 8 and later, Windows 2012, Windows 2012 R2

Symptom:

When I install a Windows boot image, I get the following error message on a “Windows Setup” message box during OS installation:

Windows could not parse or process the unattend answer file for pass [specialize]. The settings specified in the answer file cannot be applied. The error was detected while processing settings for component {Microsoft-Windows-Shell-Setup].

Solution:

This happens for one of the following two reasons:

1. Primary reason is incorrect value for "Windows TypeWindowsType" and "ProductId" boot parameters:

   For installation, "ProductId" is the valid Windows product key corresponding to the supplied "WindowsType". We support key-less installation for the Specific Editions of Windows Images such as Volume licensing image and Enterprise image. Key-less installation does not hold true for all Windows Images. For more information, see section “Windows Key-less Installation” Chapter 9 Windows OSIM OS Images.

2. Secondary reason is incorrect value for boot parameters such as "HostName", "TimeZone", and "OSUser".

To avoid this error, provide the correct value for the above mentioned boot parameters.

Chapter 16: OSIM Events

The following sections provide tables that list all events generated by the domain manager (OSIM part) for operating system installation jobs, targets, and boot server, and by the OSIM boot server.

The mapping of event ID and event meaning is done in the file ..\CA\DSM\bin\ccsmevosim.enu on the domain manager.

This section contains the following topics:

Manager Events

The following table lists all events generated by the domain manager (OSIM part) for operating system installation jobs, targets, and boot server.

Event ID	Event Meaning
2000	OSIM Manager started
2001	OSIM Manager stopped
2002	New boot server *xxx*
2003	New target computer with MAC address *xxx* reported by boot server *yyy*
2004	Target computer *xxx* attached to boot server *yyy*
2005	Target computer *xxx* detached from boot server *yyy*
2006	Image *xxx* attached to boot server *yyy*
2007	Image *xxx* detached from boot server *yyy*
2010	OSIM job *xxx* activated
2011	OSIM job *xxx* sent to boot server *yyy*
2012	OSIM job *xxx* cancel request sent to boot server *yyy*
2013	OSIM job *xxx* cancelled
2014	OSIM job *xxx* terminated with status *yyy*
2015	OSIM job *xxx* succeeded
2016	OSIM job *xxx* pending

Event ID	Event Meaning
2017	OSIM job *xxx* installing
2019	OSIM job *xxx* terminated with error *yyy*
2021	OSIM job *xxx* terminated with error: the boot configuration is not linked to a target computer
2022	OSIM job *xxx* terminated with error: the boot configuration is linked to more than one target computer.
2023	OSIM job *xxx* terminated with error: no activation time found
2024	OSIM job *xxx* terminated with error: the boot configuration is not linked to any OS image.
2025	OSIM job *xxx* terminated with error: the boot configuration is linked to more than one OS image
2026	OSIM job *xxx* terminated with error: the OS image *yyy* has no parameter values
2027	OSIM job *xxx* terminated with error: the value for parameter *yyy* was not found in the database
2028	OSIM job *xxx* terminated with error: the target computer is not assigned to any boot server
2029	OSIM job *xxx* terminated with error: the target computer is assigned to more than one boot server
2030	OSIM job *xxx* terminated with error: the boot server *yyy* has no host object
2031	OSIM job *xxx* terminated with error: communications problem: error sending message to boot server
2032	OSIM job *xxx* terminated with error: job timeout, no reply from boot server.
2033	OSIM job *xxx* terminated with error: the boot server did not accept the installation or job cancellation order
2034	OSIM job *xxx* terminated with error: the required OS image is not available on boot server *yyy*
2035	OSIM job *xxx* terminated with error: the required boot disk image *yyy* is not available on the boot server
2036	OSIM job *xxx* terminated with error: the job was forcibly aborted on user request. The installation was already started, the state of target system is unknown
2037	OSIM job *xxx* terminated with error: the 'InstallCounter' parameter is missing.

Event ID	Event Meaning
2038	OSIM job *xxx* terminated with error: the installation was aborted by the target computer. The previous installation is probably intact.

Boot Server Events

The following table lists all events generated by the OSIM boot server.

Event ID	Event Meaning
101	Boot Server started
102	Boot Server stopped
120	TFTP: File *xxx* Source *yyy*
301	MAIN: Init Error - Failed to establish TCP/IP environment - *xxx*
302	MAIN: Init Error - Failed to get host information - *xxx*
303	MAIN: Init Error - Failed to open Configuration section *xxx*. Error *yyy*
304	MAIN: Init Error - Failed to read Configuration value *xxx*. Error *yyy*
307	MAIN: Init Error - begin thread failed - *xxx*
309	REP: Init Error - Failed to open FCOR *xxx*
310	MAIN: Init Error - No valid/active interfaces were found
702	TFTP: File *xxx* Source *yyy* Socket Error - *zzz*
703	TFTP: File *xxx* Source *yyy* File open Error - *zzz*
708	TFTP: File *xxx* Source *yyy* Illegal request
709	TFTP: File *xxx* Source *yyy* File not found
710	TFTP: File *xxx* Source *yyy* No prior BINL request
803	ADS: Query *xxx* failed. Result code *yyy*
804	ADS: Failed to get Event Nofication. Result code *xxx*
806	ADS: Lost connection to the ADS provider. Result code *xxx*
901	Unable to read Configuration value *xxx*. Error *yyy*

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W