Tivoli Provisioning Manager for Images
Version 7.1.1.16

User's Guide

IBM

Tivoli Provisioning Manager for Images
Version 7.1.1.16

User's Guide

IBM

Contents

Chapter 4. Provisioning VMWare ESX Server on x86 and x86-64 targets . . . 191

Chapter 5. Provisioning VMWare ESXi on x86-64 targets. 217

Chapter 6. Provisioning non x86 and non x86-64 targets 231

Chapter 7. Managing virtual images 303

Chapter 1. Defining targets

Targets are computers known to the OS deployment server. This includes the OS deployment servers themselves, the computers on which they deploy system profiles, and reference computers from which cloned system profiles are created.

An OS deployment server must know its targets to be able to work with them. You must add any target, either automatically or manually, to an OS deployment server before it can be used. All the virtual machines that are not snapshots are discovered as hypervisors.

The Target Monitor is the main interface with your targets. You can view your targets and their status, organize them into a hierarchical structure for easy retrieval, create lists using a search function, and view them sorted by subnet. Using the Target Monitor, you can select a default administrative group into which new targets are to be attached and assigned default settings.

Target collection types

The targets known to the OS deployment server can be sorted into administrative groups, custom lists, subnets, hypervisors, or as a result of a custom dynamic search.

Administrative groups

Determine which administrators are allowed to configure which targets.

These groups can contain a hierarchy of sub-folders.Every target belongs to exactly one administrative group.

One administrative group is the *default* group that registers unknown targets when they first contact the OS deployment server.

Note: Options defined for unknown targets might not be identical to those defined for the *default* group. An unknown target boots the first time using the options set for the unknown targets. After it is registered in the default group, it uses the options set for computers in this group for subsequent boots.

Custom target lists

Custom target lists are groupings of targets built by system administrators to run tasks on several targets together. A single target can belong to several custom target lists.

These groups can contain a hierarchy of sub-folders.You can build a custom target list by adding individual targets one at a time, or through a search query. Launch this search query through the **Create a custom target list from a search query** option that appears when you select a custom target list folder.

Subnets

Subnets group targets according to their IP address. A target can only belong to one subnet at a time. Multi-homed targets are listed in the subnet on which they last made a network-boot.

Hypervisors

Hypervisors group virtual targets discovered by **`rbagent host inventory`** and listed in the image monitor, according to their hypervisor. A target can only belong to one hypervisor at a time.

Dynamic searches
With dynamic searches, you can customize the selection of targets to be displayed in a group. To start the wizard, right-click the **dynamich searches** folder and choose **Add dynamic search**. In the wizard, you can use one of the listed search parameters or, if they are not right for the search you need, you can use the **Free text condition** field.

The **Free text condition** in dynamic searches completes the SQL query made against the BOM table in the database. For example, if you want to find all targets whose MAC address starts with `00:0C:29`, you can use the following syntax in **Free text condition**:

```
MAC LIKE '00:0C:29:%'
```

The **BiosVersion** field is not contained in the BOM table but, you can extend the search to other tables in the database using a nested query and a common field in the BOM table and in another table that defines a target (such as UUID).

If you want to identify the target with a specific Bios Version, the **BiosVersion** value is contained in the DMIInventory table. So, if you want to search all targets having 6.00 as Bios Version, enter the following in the **Free text condition** field:

```
UUID in (select UUID from DMIInventory where BiosVersion='6.00')
```

In the same way, you can extend the search to all other fields contained in DMIInventory, where the inventory data of the targets is stored, for example: ProcCount to have a list of the targets with a certain amount of processors. So, to see the list of targets with two processors or more, enter the following free text condition:

```
UUID in (select UUID from DMIInventory where ProcCount>=2)
```

Target information

The following target information is readily visible in the Target Monitor:

IP address
The target IP address

arch The target platform, for example, Intel, or Oracle

model The computer model of the target

serial The serial number of the target

? The state of Tivoli® Provisioning Manager for Images, illustrated with icons

updated
The last time that the state information was updated

status The last deployment status of the target

Note: You can modify how the information columns are displayed in the Target Monitor by clicking **Arrange columns** in the contextual menu. You can customize the size of the columns, their relative order, and which columns are displayed.

Adding targets

You must add and configure a target before you can start a deployment for it.

The examples for preparing targets are based on the deployment of one target; the process for deploying multiple targets is similar.

Ensure that you have at least one OS configuration to deploy. Having one or two software modules ready makes the deployment more useful but is not mandatory.

The Target Monitor is used to manage and deploy targets. To access the Target Monitor, click the first item of the menu in the web interface.

Methods for adding targets

Select the target on which you want to deploy the OS configuration you have created. To start the deployment, the target must be visible in the Target Monitor. There are several ways to make the target appear in the Target Monitor. In all cases, it is important to configure your target to start on the network, or to press the network boot hot-key, for example, `F12`, when the target starts. You can add the target into the Target Monitor in the following ways:

- Let the Target Monitor detect the target. The target is started and boots on the network. The target is displayed in the targets tree on the Target Monitor page, if the OS deployment server is not running as a *closed server*.
- Create the target manually. The target must be identified by either its MAC address, IP address, Unique Universal Identifier (UUID), or serial number.
- Use a target list. A target list file is a `.csv` text file with comma separated values. Lists for targets are useful for adding large numbers of targets to the OS deployment server without having to start them up individually on the network.

Detecting targets automatically

You can detect targets automatically.

The OS deployment server is configured to automatically answer every PXE target that requests a network boot program. Any known PXE target is added to the target database.

1. Turn on your target and make it start on the network. At this stage, the target appears in the Target Monitor, in the target tree.
2. Select the default group usually called **Default**. You see an icon representing the target that you have started.

If no OS configuration is bound to the target, the target shows a locked screen.

If the computer you have just booted was used to create an OS configuration or was used in a previous deployment, the locked screen might be skipped and a menu with bound OS configurations displayed instead. This happens because OS configurations are already bound to the target that you are starting.

In a network with several PXE servers

In an environment with multiple PXE servers, the easiest method for populating the target database is to:

1. Stop all PXE servers except for the OS deployment server.
2. Boot PXE targets that must be inserted in the OS deployment server database.
3. Restart the PXE servers and set OS deployment server to ignore new targets.

Adding targets to the Target Monitor manually

If you want to perform a deployment without having to start targets first, you can add targets manually into the Target Monitor or import a comma-separated text file containing a list of targets to be added.

1. Go to the **Target Monitor** page on the web interface of the OS deployment server.
2. Select either an administrative group or the **by Administrative group** folder. New targets are always inserted within an administrative group.
3. Click **Register new targets**.
4. In the window, enter at least one of the following target identifiers:
   - **MAC address**
   - **IP address**
   - **Serial number**
   - **UUID**
   - **Hostname**

   Note: The **IP address** and **Hostname** are required to deploy targets other than x86 and x86-64.

   When deploying Linux on PowerPC® and Cell Blades, a default **Hostname** is provided if none was registered.
5. Click **Ok**.
6. Add another target or click **Cancel** to close the **Register target** window.

When the target is added to the database, it appears in the target tree.

Note: If you have entered a wrong identifier for a target, and you want to remove that target from the Target Monitor and from the database, right-click the target and select **Delete** from the contextual menu.

Importing and exporting targets lists

A target list file is a `.csv` text file with comma-separated values. Importing a target list is useful for adding large numbers of targets to the OS deployment server without having to start them individually on the network. You can also import a PCI inventory for a single target in an `.ini` file.

Familiarize yourself with target lists and PCI inventory.

Target list

Before you can import a target list, you must either export one or create a new one.

Information about each target in a target list is a collection of more than seventy items, including:

- MAC address
- IP address
- User parameters
- Motherboard information
- Processor information

To view the complete list of items, export a target list and read the beginning of the `.csv` file.

For the OS deployment server to successfully import targets in a list, you must fill in at least one of the following items:

- Serial number
- MAC address
- UUID
- IP address

The item you fill in can vary from target to target. Other items can remain empty.

Target lists above 1 GB in size cannot be imported into an OS deployment server because of browser limitations. Therefore, you cannot use target lists for more than about 1000 targets.

Note: Do not use target lists to back up target information. To back up target information, you must back up the database with an appropriate tool. Lists of targets are not as complete as the database. In particular, target lists do not include some crucial target information found in the database, for example:

- Bindings
- Disk inventory
- PCI inventory
- Deployment history

PCI inventory

You can export a PCI inventory to a USB key or disk. They can be booted through a network boot media, without having network drivers.

- **Importing a target list**
  1. Go to the**Target Monitor page** in the web interface.
  2. Click **Import targets**.
  3. Indicate the location of the `.csv` file.
  4. Click **Ok**.
- **Exporting a target list**
  1. Go to the**Target Monitor** page in the web interface.
  2. Click **Export targets**.
  3. Click **Save**. You can change the default file name, `hostexport.csv`, and location.
- **Importing a PCI inventory**
  1. Go to the**Target Monitor page** in the web interface.
  2. Click **Import targets**.
  3. Indicate the location of the `newhost.ini` file.
  4. Click **Ok**.

Configuring new targets

Targets are assigned default parameters at the time they are added to the OS deployment server. You can configure new targets by changing these default parameters.

You can configure how the OS deployment server accepts targets attempting to boot, in which group they are added, which Tivoli Provisioning Manager for Images kernel options to use, and whether to enable human interfaces.

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select **Idle Layout** and then **Idle state**.
3. Click **View idle parameters**.
4. Click **Edit** in the **Handling of unknown targets** banner. This opens the **Handling of unknown targets** dialog
5. Modify the parameters you need. You can also decide not to include targets to the OS deployment server.
6. Click **OK**.

Targets that are added to the OS deployment server are now assigned these new default parameters.

Setting the type of target boot

You can set the type of boot of a target or a group of targets, depending on the task.

By default, targets boot in kernel mode. For Unified Extensible Firmware Interface (UEFI) targets, you must change the target boot to the kernel-free mode. It is optional to change it for BIOS targets. You can do this in the following ways:

From the hardware configuration rule

You can force the boot to the kernel-free mode by creating a rule that specifies a pattern describing the target computer models to which the rule applies, for example `VMWare*`, and the flag to enable kernel-free boot. .

When a target boots for the first time from an OS deployment server and its model is unknown to the OS deployment server, the hardware configuration rule is not applied. The first boot occurs according to the server configuration. After an inventory is run on the target and its model its discovered, target boots are run according to the hardware configuration rule settings. To force the first target boot from the hardware configuration rule, manually define the target in the Target Monitor page, specifying its model. Now, when the target performs the first boot, its model is already known and the hardware configuration rule is applied

To run a target boot in kernel-free mode, create a new hardware configuration rule or change the boot options of an existing rule as follows:

1. Go to **Server** > **Advanced features** > **Hardware configurations**.
2. Select the rule and click **Edit rule**.
3. In the **Boot engine options** section, select **Use kernel-free mode**.
4. Complete the wizard.

From the server configuration

This option is used at first discovery of the target. After the target boots, the flag is set on the target-specific configuration. If you choose the kernel-free deployment mode, only these options are available: **Reboot on fatal errors**, **Do not reboot on errors**, and **Disable multicast**.

If a target previously unknown to OS deployment server performs a UEFI PXE boot, the target boot setting **`Use kernel-free mode`** is set to **`Yes`** to complete the UEFI PXE boot, regardless of the corresponding server-wide setting. If you change the boot setting manually, it is not reset at subsequent PXE boots.

To run a target boot in kernel-free mode, update the server configuration stored in the `rembo.conf` file as follows:

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select **Idle Layout** and **Idle state** folder.

   Note: If you have a multiserver hierarchy and you perform changes in the **Idle state** layout of a parent server, when you replicate from one server to another, the **Idle state** changes are not replicated. Parameters with a red dot are not replicated from one server to another because these are server specific parameters.
3. Click **View idle parameters**.
4. Click **Edit** in the **Handling of unknown targets** banner.
5. Select **Use kernel-free flow**. The kernel-free mode is set for all the targets that boot on this server.

From the target configuration

This option is set to change the default mode. It is overridden by the hardware rule, if **Use kernel-free flow** is set.

To run a target boot in kernel-free mode, update the target configuration as follows:

1. Select a single target or multiple targets on the Target Monitor page. To do this, go to **Server** > **OS deployment** > **Target Monitor**. . To select multiple targets or deployment, select an **administrative group**, **custom list**, **subnet**, or click **individual target** names while holding down the Ctrl key.
2. Select **Edit boot settings**.
3. Select **Use kernel-free flow**.
4. Complete the wizard.

Changing the default administrative group

You can change the administrative group to which new targets are automatically assigned. By default, new targets are assigned to the *Default* group.

To see the current default administrative group to which new targets are assigned, select any administrative group. Information is provided. To see this information, go to **Server** > **OS deployment** > **Target Monitor**. .

To change the default administrative group:

1. Optionally, create a new administrative group.
   a. Select **by Administrative group** in the Target Monitor
   b. Click **Add a new admin** in the contextual menu to create a new administrative group.
2. Go to **Server** > **OS deployment** > **Task templates**.
3. Select **Idle Layout** and **Idle state**.
4. Click **View idle parameters**.
5. Click **Edit** in the **Handling of unknown targets** banner.
6. In the second section of the new dialog, select the new default administrative group from the list.

Requirements for VMware targets

To deploy system profiles on VMware, your system must meet a number of requirements when you set up the VMware target.

Guest operating system

Set the guest operating system type and the version that you are deploying on the target.

Network adapter

- Windows The Intel e1000 network adapter works correctly on all Windows editions
- Windows On Windows 64-bit, the AMD Lance network adapter is not supported. Using it results in a failed deployment with either a shutdown of the virtual machine or a stop error on a blue screen.
- Linux The AMD Lance network adapter is supported for all Linux distributions, but is slow.
- Linux The Intel e1000 network adapter is supported on all Linux distributions, except for Red Hat Enterprise Linux (REHL5).

  With REHL5, the Intel e1000 card is not functionable when rebooting the operating system after performing Linprep. The target can no longer connect to the network and the deployment stops and fails. To work around this issue, install two network cards on your VMware target:

  - The Intel e1000 as the primary boot device
  - An AMD Lance as the second boot device to use as a fallback.

  With the two cards, when Linux reboots and the Intel e1000 does not answer, the AMD Lance takes over, allowing the virtual machine boot and the deployment continue.
- SUSE For cloning and Direct Migration of SuSE Linux Enterprise Server, you must use the Intel e1000 network adapter.

SCSI controller

Paravirtual SCSI controllers are not supported.The compatibility between SCSI controllers and Windows operating systems on VMware targets is described in Table 1.

Table 1. Compatibility between Windows operating systems and SCSI controllers

Compatibility	Windows XP	Windows 2003/Vista/2008/ 7/8/2012	WinPE 3.x	WinPE 4.x
BusLogic	Yes	No	No	No
LSI Logic	No	Yes	Yes	Yes

- To deploy Windows 2003/Vista/2008/7, use the LSI Logic driver.
- To deploy Windows XP, you have the following options:
  - Before installing a Windows XP guest operating systems on a VMWare hypervisor, with BUSLogic adapter, you must inject it offline into a WinPE deployment engine, depending on your VMware version.
  - You can make a software module with the LSI Logic Parallel driver, and bind it to your Windows XP system profiles
- On all other operating systems, LSI Logic is supported.

Note: LSI Logic driver for Windows Server 2003 operating system, `symmpi.inf`, version 1.28.03 is supported.

Injecting drivers on WinPE to deploy Windows XP guests

Before installing a Windows XP or Windows XP guest operating systems on a VMWare hypervisor, with BUSLogic adapter on VMWare, you must inject the VMWare SCSIAdapter BusLogic drivers into the standard WinPE. These drivers are not contained on the Vista 7 DVD or installed with VMWare tools.

- Create a virtual machine and ensure that its devices are set up correctly.
- On Tivoli Provisioning Manager for Images, create the WinPE deployment engine to contain the necessary BusLogic drivers. Assign relevant matching models to this WinPE deployment engine, for example `*VMware*4.1*`.
- On VMware 4.1, disable any virus scan, to improve performance in the WinPE update.

If you use VMware 3.5, you can user dynamic driver injection and bind your driver software modules to the WinPE deployment engine using the driver binding grid. If you use VMware 4.1, you must inject the VMware missing drivers offline in the WinPE deployment engine.

1. Extract the Microsoft drivers needed to run a Windows virtual machine on VMware on a virtual USB key or disk.
   a. Install a Windows virtual machine on VMware.
   b. From the VMware Workstation menu, select **VM > Install VMware Tools**, The VMware Workstation connects the virtual machine disk drive to the ISO image file that contains the VMware Tools installer for your guest operating system. After the installation process, a new disk is bound to VMware and you can see all the required drivers.
2. Create software modules for the newly extracted drivers.
3.
   - With VMware 3.5, bind your newly created driver software modules to the WinPE deployment engine.
   - With VMware 4.1, inject the driver software modules offline into an existing WinPE deployment engine.
     a. Go to **Server** > **Advanced features** > **Deployment engines**.
     b. Double-click the name of a deployment engine to view its details.
     c. Select **Inject driver** in the contextual menu.
     d. In the wizard, specify a computer running the web interface extension.
     e. Select the driver software modules to inject in the WinPE deployment engine.

        Note: Injected drivers cannot be removed from WinPE. These drivers are started regardless of whether they are compatible with the hardware.
     f. Complete the wizard.

Your BusLogic driver is now either bound to, or contained in, your WinPE deployment engine.

You can now install Windows XP with the WinPE deployment engine on VMware, and then use your guest target like any other virtual machine.

Booting non-x86 and non-x86-64 targets

You can boot targets that do not have an x86 or an x86-64 architecture.

Booting pSeries targets on the OS deployment server

You can boot pSeries machines on the OS deployment server.

Before you can boot a pSeries target on the OS deployment server, you must:

- Verify the network connectivity as follows:
  1. From the SMS menu, test the network interfaces using the **Setup Remote IPL (Initial Program Load)** menu.
  2. Select the interface to use for the deploy.
  3. Configure it and run a ping test to verify the connectivity.

     Note: Ensure that the selected interface is recognized by the operating system during the installation phase.
- Manually register the pSeries target in the OS deployment server, indicating at least the MAC address and the host name.
- Run the **devalias** command to select the correct boot interface and add it.
- Configure the TCP/IP options.
- Start a deployment task on the target. Without a task bound to it, the target cannot boot on the OS deployment server.

How you boot a pSeries target on the OS deployment server depends on the operating system you want to install.

- AIX SUSE To install AIX® and SuSE 10:
  1. Boot the target using the **boot net** command.
  2. Type 1 to select **SMS Menu**.
  3. Type 5 for **Select Boot options**.
  4. Type 1 for **Select Install/Boot Device**.
  5. Type 6 for **Network**.
  6. Under **Select device**, select the network interface that you have registered in the OS deployment server. If you are not booting from the default network interface, use the alias of the interface instead of the PCI identifier.
  7. Type 2 for **Normal Mode Boot**.
  8. Type 1 (Yes) to confirm the above.

     Note: If the standard Linux operating system booting stops and you are using the serial console access, to solve the problem press any key in the `Autoyast` boot prompt. Type **linux console=hvsi0** and press enter.
- Red Hat To install RedHat
  1. Before booting ensure that you are using the standard network card, otherwise perform the following steps:
     - – Switch to the OpenFirmware prompt and list the boot aliases using the **devalias** command.
     - – If the interface from which you are going to boot is listed in the aliases you can continue. If the interface is not included in the devalias list, create a new alias. Run **ls** to list all the devices and see the device address of the network card.

- Add a new alias using **devalias** such as: `devalias net2 /pci@800000020000203/ethernet@1`

2. Boot the target using the **boot net** command.
3. Press 8 when booting to reach the Open Firmware prompt.
4. From an Open Firmware prompt, run `boot net ks=http://serverip:serverport/linux/ks.cfg ksdevice=eth0`. `serverip` is the IP address of the OS deployment server, and `serverport` its port. Serverport is typically 8080. To boot from a different network card use the alias previously defined: `boot net2 ks=http://serverip:serverport/linux/ks.cfg ksdevice=eth2` The chosen interface is recognized as `eth2` during the operating system installation.

Booting CellBlades targets on the OS deployment server

You can boot CellBlades on the OS deployment server.

To boot on the OS deployment server, perform the following steps:

1. Boot the target using the **boot net** command.
2. Press 8 when booting to reach the Open Firmware prompt.
3. From an Open Firmware prompt, run `boot net ks=http://serverip:serverport/linux/ks.cfg ksdevice=eth0`. `serverip` is the IP address of the OS deployment server, and `serverport` its port. Serverport is typically 8080.

If the server IP is 192.168.1.25, and the server HTTP port is 8080, in the Open Firmware prompt, enter: `boot net ks=http://192.168.1.25:8080/linux/ks.cfg ksdevice=eth0`

Booting SPARC targets on the OS deployment server

Booting SPARC targets on the OS deployment server requires a few prerequisites and depends on whether you are doing it from OpenBoot or from a running operating system.

Ensure that you have the following setup:

- DHCP option 66 must be set to the IP address of the OS deployment server.
- DHCP option 67 must be set to `rembo.fcode`.
- To network boot with Tivoli Provisioning Manager for Images, the SUN SPARC target must support WAN boot. The Open Boot version of the SPARC target must be equal or greater than 4.17.1. To verify if a SPARC target running under Solaris supports WAN boot run the following command:

  ```
  # eeprom | grep network-boot-arguments
  ```

  If the variable `network-boot-arguments` is displayed, or if the previous command returns the output `network-boot-arguments: data not available`, the OBP supports WAN boot installations. Do not update the OBP before you perform your WAN boot installation. If the previous command does not return any output, the OBP does not support WAN boot installations. You must perform one of the following tasks.

  - Update the target OBP. See your system documentation for information about how to update the OBP.
  - After you complete the preparation tasks and are ready to install the target, perform the WAN boot installation from the Solaris Software media in a local drive.

- You must register your SPARC target on the OS deployment server by indicating at least its IP address and its host name.
- Network boot on the OS deployment server for Solaris is accepted only when a deployment task is scheduled on that target .

You can boot a SPARC target on the OS deployment server either when the target is booting, or when the Solaris operating system is running. You can also use a dynamic or a static IP address.

- From the OpenBOOT monitor (Stop-A), type `boot net:dhcp`. To make this change permanent, type `setenv boot-device net:dhcp`. Then a simple `boot` command or a cold boot are enough to boot onto the OS deployment server. If `setenv boot-device net:dhcp` does not work, use a static IP address.
- To boot with a dynamic IP address from the OpenBOOT monitor (Stop-A), type

  ```
  setenv network-boot-arguments dhcp,file=http://<OSDeploymentServerIP>:8080/sun4u
  boot net - install
  ```

  where <OSDeploymentServerIP> is the IP address of your OS deployment server.
- To boot with a static IP address from the OpenBOOT monitor (Stop-A), type

  ```
  setenv network-boot-arguments host-ip=<client-IP>,router-ip=<router-ip>,
  subnet-mask=<mask-value>,file=http://<OSDeploymentServerIP>:8080/sun4u
  boot net - install
  ```

  where:

  <client-IP>
  Is the IP address of the target.

  <router-ip>
  Is the IP address of the router.

  <mask-value>
  Is the subnet mask value.

  <OSDeploymentServerIP>
  Is the IP address of your OS deployment server.
- To force a network boot from the operating system, use

  ```
      /usr/platform/sun4u/sbin/eeprom boot-device=net:dhcp
      /usr/sbin/reboot
  ```

  Alternatively, you can force a single network boot by using the following special string, that is recognized by the bootstrap code of the OS deployment server

  ```
      /usr/platform/sun4u/sbin/eeprom boot-device="net:dhcp was: disk"
      /usr/sbin/reboot
  ```

  Note: For architectures other than `sun4u`, change the path above. Use the `uname -m` command to check the architecture.
- If you are running the web interface extension as a service on a SUN target , you can use the Target Monitor option to automatically reboot the target from the web interface. This generates the one-time change of boot device described above.

Booting on UEFI targets

Understand the partition layout used by Unified Extensible Firmware Interface (UEFI) targets.

You can boot UEFI targets on the OS deployment server.

UEFI targets use GUID Partition Table (GPT) partition layout instead of Master Boot Record (MBR). The GPT partition layout uses the EFI System Partition (ESP) and Microsoft Reserved (MSR) partition types. If no ESP or MSR partition is specified on the target that you are deploying, the ESP and MSR are automatically added in real time.

ESP is used for both Linux and Windows targets, while MSR is used for Windows only.

Organizing targets

Targets in the Target Monitor are organized into administrative groups, custom lists, and subnets.

An administrative group has a hierarchical tree structure, and can be used by system administrators to grant or deny access to specific web interface operators to configure particular target groups.

The administrative group is the unique classification for each host in the database. A host can appear only one time in the tree. Custom lists are lists built by system administrators to run tasks on several targets at the same time. You can build a custom target list by adding individual targets, or as the result of a search query.

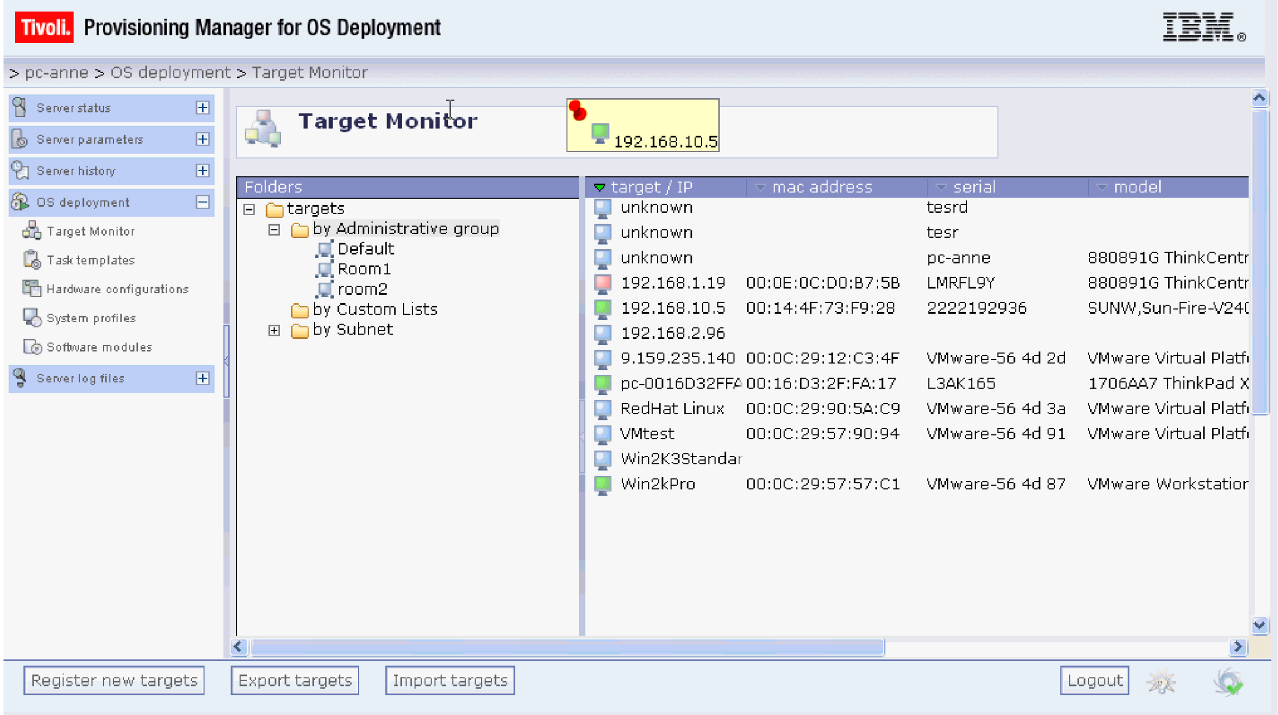
In the tree by `Custom list`, each node is a list based on a queries result. A host can be a member of more than one list. Exporting a custom list generates a CSV file with all the properties of the selected hosts, so you find the name of the administrative group on which the host is member but not the name of the custom list, of which there might be more than one. This is why a custom list is not a significant parameter for a host.

Subnets implicitly and automatically group targets according to their IP address. Multi-homed targets (with more than one network interface) are listed as part of the subnet on which they last made a network-boot. Subnets cannot be modified by the users.

When a new computer is added to the database, either manually or because the target was started in network boot mode, the Target Monitor automatically places this target in the *default* administrative group. To see which group is the default administrative group, select any administrative group: information is displayed below the target tree. The name of the default administrative group is listed.

You can move targets from one group or custom list to another:

Drag-and-drop the icon from one group or custom list to another. You might want to use the pin-board in the web interface title bar, for example if the destination folder is not visible. You can temporarily leave the dragged target on the pin-board while you search for and open the correct folder.Figure 1 on page 14 illustrates this process.



target / IP	mac address	serial	model
unknown		tesrd	
unknown		tesr	
unknown		pc-anne	880891G ThinkCentr
192.168.1.19	00:0E:0C:D0:B7:5B	LMRFL9Y	880891G ThinkCentr
192.168.10.5	00:14:4F:73:F9:28	2222192936	SUNW,Sun-Fire-V240
192.168.2.96			
9.159.235.140	00:0C:29:12:C3:4F	VMware-56 4d 2d	VMware Virtual Platf
pc-0016D32FFA	00:16:D3:2F:FA:17	L3AK165	1706AA7 ThinkPad X
RedHat Linux	00:0C:29:90:5A:C9	VMware-56 4d 3a	VMware Virtual Platf
VMtest	00:0C:29:57:90:94	VMware-56 4d 91	VMware Virtual Platf
Win2K3Standar			
Win2kPro	00:0C:29:57:57:C1	VMware-56 4d 87	VMware Workstation

Figure 1. Pin-board of the Target Monitor

Configuring targets

You can configure targets individually or together, using administrative groups, custom lists, subnets, or multiple selection.

To configure a single target:

1. Select an individual target.
2. Click **View target details** in the contextual menu.
3. Click **Edit** in the banner of each group of parameters that you want to modify.

Configuring multiple targets

1. Select multiple targets, an administrative group, a custom list, or a subnet.
   - If you have selected multiple targets, edit links for each group of parameters appears at the bottom of the Target Monitor.
   - If you have selected an administrative group, a custom list, or a subnet, click **Edit targets in list** in the contextual menu. The **Target multi-edition** window appears.
2. Click the edit link for each group of parameters.
3. Select the options that you need. You can view and select sub-options.
4. Click **Save** to close **Target multi-edition** and return to the **Target Monitor**.

Configuring targets for fully unattended OS deployments

To run fully unattended deployments, you must configure some parameters depending on the operating system to be deployed. Configuring targets is an alternative to providing the data in OS configurations.

You must configure your target before you start a deployment. Some values are mandatory for a fully unattended deployment and must be filled in at the target level if the information is not included in the OS configuration.

Note: If multiple targets share the same information, you can set fixed values in the OS configuration that you are deploying on these targets. Fixed values at the OS configuration level override values entered in the **Target details** page, and are used by all the targets deploying the OS configuration containing the fixed values.

1. Double-click the target to access its details page. The **Target details** page contains all of the properties specific to this target, including the target name, serial number, and the product key to use when installing an operating system.

   Windows **For Windows deployments**

   The following fields are required by Sysprep::
   - Target name
   - Product key. You can copy and paste the key in xxxxx-xxxxx-xxxxx-xxxxx-xxxxx format, unless you are deploying Windows Vista/2008/7/2012 with a Volume License.
   - User full name and organization
   - Administrator password
   - Workgroup or domain name

   Solaris **For Solaris deployments**
   - Solaris standard installation procedure includes checking for a valid computer name and IP that matches DNS and DHCP.
   - The Solaris NFS server must have name resolution configured to recognize the target name.
   - For each of the name resolution methods in Solaris, You must set a specific set of fixed properties.

     DNS
     - At least one DNS server
     - A DNS domain
     - A DNS domain search order

     NIS and NIS+
     - A DNS domain
     - A NIS name server

     LDAP
     - A DNS domain
     - An LDAP name server
     - An LDAP profile
   - The OS deployment server uses the root user information provided in the target specific details, or profile details, during installation. If this information is not configured, the default value of root user password is ""
2. Enter the mandatory fields and click **OK** to validate your changes.
3. If you have used the Target Monitor on this target before and the target is not displaying the *locked screen*, you might want to remove the OS configuration bindings that are forcing it into specific OS configurations.
   a. Double-click your target.

b. Select the **OS configurations** tab.
c. Click **Edit**.
d. Clear the items and click **OK** to remove the bindings.

Setting partition sizes on the targets

If you need different partition sizes on your targets, but you want to deploy them with the same system profile, you can set the partition size by target.

1. Edit the **User details** section of the **Target details**.
2. In **User Category 9**, enter the partition size information. Use the following syntax:**`resize [ <existing mount point> <size in MB>]`**.

   For example, type `resize c 5000 d 10000`.

   Note: This information overrides the partition size information in the OS configuration and in the system profile.
3. Click **Save**.

The next time you deploy this target, the partitions that you have specified are resized according to the values you provided.

Chapter 2. Provisioning Windows operating systems on x86 and x86-64 targets

You can deploy Windows operating systems on targets using Windows Preinstallation Environment (WinPE) deployment engines.

Overview of WinPE deployment engines

WinPE deployment engines are a prerequisite for provisioning Windows operating systems.

Windows Preinstallation Environment (WinPE) is a group of files that can be loaded as a ramdisk so that you can perform operations on a target. Without WinPE, you cannot provision Windows operating systems. There are several advantages to using a WinPE deployment engine:

- It has a small footprint.
- The memory usage is at a minimum, creating an optimization in the ramdisk boot.
- It contains more built-in drivers.

WinPE deployment engines are stored under **Server** > **Advanced features** > **Deployment engines**.

WinPE versions

The only compatible versions for the WinPE deployment engine are 3.x, 4.x, and 5.x, depending on the tasks to be performed.

WinPE 3.x must be created from a Windows Automated Installation Kit (AIK) for Windows 7 in English.

WinPE 4.x must be created from a Windows Assessment and Deployment Kit (ADK) for Windows 8 in English.

WinPE 5.x must be created from a Windows Assessment and Deployment Kit (ADK) for Windows 8.1 in English.

Note: WinPE 3.x and WinPE 4.x deployment engines can coexist on the same OS deployment server and can be bounded both to the same machine models. Selection of WinPE 3.x or WinPE 4.x is automatically made during deployment depending on the Windows version to deploy and to the matching model. WinPE 4.x is used to create and deploy Windows 8 and Windows 2012 profiles. WinPE 5.x is used to deploy Windows 8.1 and Windows 2012 R2 profiles.

WinPE 32-bit and WinPE 64-bit deployment engines

WinPE 3.x deployment engines are extracted from Windows AIK, one 32-bit version and one 64-bit version.

WinPE 4.x deployment engines are extracted from Windows ADK, one 32-bit version and one 64-bit version.

In the current version of the product, the 32-bit WinPE deployment engine is used for all the tasks requiring a WinPE deployment engine. In BIOS mode, the 64-bit WinPE 3.x is used only to deploy Windows Vista 64-bit and Windows 2008 64-bit unattended setup system profiles. For these operating systems, both versions of WinPE 3.x deployment engine are used together. In UEFI mode, the 64-bit WinPE is used for all the tasks requiring a WinPE deployment engine.

Target operating systems and component combinations

The following table lists, for each supported operating system, the component combinations you can use for re-imaging and bare metal deployments.

Table 2. Target operating systems and component combinations for re-imaging and bare metal provisioning

WIM Toolkit	Operating System
WADK 8.1 *(WinPE 5)*	Windows 8.1
	Windows Server 2012 R2
WADK 8 *(WinPE 4)*	Windows 8
	Windows Server 2012
WAIK *(WinPE 3)*	Windows 7
	Windows Vista
	Windows XP
	Windows Server 2008 R2
	Windows Server 2008
	Windows Server 2003

Notes

1. Direct re-imaging from Windows XP or Windows Vista to Windows 8.1 is not supported. It must done as a two-step process. You must first re-image the target to Windows 7 or Windows 8, and subsequently re-image to Windows 8.1.
2. When re-imaging from Windows XP to Windows 7, WinPE5 is not supported.

Note: MDT 2013 requires WADK 8.1.

Deployment engine creation

If your OS deployment server runs on a Windows operating system and if you have Windows AIK installed on the server, then when it starts up, the OS deployment server checks that there is a WinPE deployment engine on the server. If not, it creates it automatically. The process takes several minutes and you cannot log in to the web interface during the process.

If your OS deployment server does not run on a Windows operating system, or if you want additional WinPE deployment engines, you can create them manually.

Working with several WinPE deployment engines

In most cases, you do not need to create additional WinPE deployment engines, because one per architecture is enough most of the time.

When the WinPE deployment engine is transferred to a target, for example, during a deployment, it contains all the drivers that are bound to this deployment engine, even if only those bound for the specific target model are used. If you are binding many drivers to account for a very large range of hardware, the size of your WinPE deployment engine might become too large for some targets that have a small RAM. In this case, you might want to create an additional WinPE deployment engine, match it only to the target with the small RAM, and bind only the drivers needed for this specific target. The size of the new WinPE deployment engine transferred to the target is much smaller.

When you have several WinPE deployment engines for the same computer architecture, you must make sure that you have specified matching model patterns that allow the OS deployment server to dispatch the WinPE deployment engines to the correct targets.

The method uses the Microsoft **drvload** command to inject drivers. If this command does not work, you must inject the drivers in the standard way.

Windows Automated Installation Kit

You need Windows Automated Installation Kit (AIK) to perform different tasks when you are provisioning Windows operating systems.

You need Windows AIK to:

- Create a WinPE deployment engine
- Create an unattended setup system profile of a Windows Vista/2008/7 operating system
- Create a cloning system profile from a Windows WIM image
- Create a Windows PE-based network boot CD/DVD

Installing Windows AIK

The current version is Windows AIK for Windows 7 in English.

Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

Best practices

Given the numerous uses of Windows AIK in the process of provisioning Windows operating systems, it is a good practice to perform all these tasks on one system that is installed with all the requirements. This target must have:

- A Windows XP/2003/Vista/2008/7 operating system
- Windows AIK for Windows 7 in English installed
- The web interface extension installed and started with local administrator privileges

If your OS deployment server is on a Windows operating system, you can use your OS deployment server as the dedicated Windows system.

Checking the version of Windows AIK

If you are unsure of the version of Windows AIK installed on a system, you can verify it.

- On Windows XP and Windows 2003:
  1. In the **Control Panel**, select **Add or Remove Programs**.
  2. Select **Windows Automated Installation Kit**.
  3. Click **Click here for support information**.
  4. Check that the version number is 2.0.0.0, which corresponds to Windows Automated Installation Kit (AIK) for Windows 7 in English.
- On Windows Vista and Windows 2008:
  1. In the **Control Panel**, if you are in the **Control Panel Home** view, select **Programs**, otherwise skip this step.
  2. Select **Programs and Features**.
  3. Select **Windows Automated Installation Kit**.
  4. If you cannot view the version number in the selected line, you can add a column with this information:
     a. Select **View** and then **Choose Details...**.
     b. Select **Version** and click **OK**.
  5. Check that the version number is 2.0.0.0, which corresponds to Windows Automated Installation Kit (AIK) for Windows 7 in English.
- On Windows 2008 R2 and Windows 7:
  1. In the **Control Panel**, if you are in the **Control Panel Home** view, select **Programs**, otherwise skip this step.
  2. Select **Programs and Features**.
  3. Select **Windows Automated Installation Kit**.
  4. If you cannot view the version number at the bottom of the screen, select **Organize** > **Layout** > **Details pane**.
  5. Check that the version number is 2.0.0.0, which corresponds to Windows Automated Installation Kit (AIK) for Windows 7 in English.

Windows Assessment and Deployment Kit

You need Windows Assessment and Deployment Kit (ADK) to perform different tasks when you are provisioning Windows 8 and Windows 2012 operating systems.

You need Windows ADK to:

- Create a WinPE deployment engine
- Create an unattended setup system profile of a Windows 8/2012 operating system
- Create a cloning system profile from a Windows WIM image
- Create a Windows PE-based network boot CD/DVD

Installing Windows ADK

The current version is Windows ADK for Windows 8 in English language.

Windows Assessment and Deployment Kit (ADK) for Windows 8 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

You must install the following required Windows ADK components:

- Windows Preinstallation Environment (Windows PE)
- Deployment Tools

Best practices

Given the numerous uses of Windows ADK in the process of provisioning Windows operating systems, it is a good practice to perform all these tasks on one system that is installed with all the requirements. This target must have:

- A Windows Vista/2008/7/8/2012 operating system
- Windows ADK for Windows 8 in English installed
- The web interface extension installed and started with local administrator privileges

If your OS deployment server is on a Windows operating system, you can use your OS deployment server as the dedicated Windows system.

Checking the version of Windows ADK

If you are unsure of the version of Windows ADK installed on a system, you can verify it.

- On Windows Vista and Windows 2008:
  1. In the **Control Panel**, if you are in the **Control Panel Home** view, select **Programs**, otherwise skip this step.
  2. Select **Programs and Features**.
  3. Select **Assessment and Deployment Kit**.
  4. If you cannot view the version number in the selected line, you can add a column with this information:
     a. Select **View** and then **Choose Details...**.
     b. Select **Version** and click **OK**.
  5. Check that the version number is 8.59.25584, which corresponds to Windows Assessment and Deployment Kit (ADK) for Windows 8 in English.
- On Windows 2008 R2, Windows 7, Windows 8, Windows 2012:
  1. In the **Control Panel**, if you are in the **Control Panel Home** view, select **Programs**, otherwise skip this step.
  2. Select **Programs and Features**.
  3. Select **Assessment and Deployment Kit**.
  4. If you cannot view the version number at the bottom of the screen, select **Organize** > **Layout** > **Details pane**.
  5. Check that the version number is 8.59.25584, which corresponds to Windows Assessment and Deployment Kit (ADK) for Windows 8 in English.

Creating aWinPE 3.x deployment engine

To create or deploy Windows profiles, you must have created a WinPE 3.x deployment engine.

Ensure that the computer from which you create the WinPE 3.x deployment engine satisfies these conditions:

- Runs a Windows operating system
- Has Windows Automated Installation Kit (AIK) for Windows 7 in English installed. Windows Automated Installation Kit (AIK) for Windows 7 in English

is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

- Runs the appropriate web interface extension (rbagent). If the Windows operating system is 64-bit, stop the 32-bit web interface extension and start the 64-bit web interface extension as follows:

  ```
  C:\TPMfOS Files\global\http\rbagent64.exe -d -v 4 -s <IPServer>:<PasswordServer>
  ```

  where:

  <IPServer>
  Specifies the IP address of the OS deployment server

  <PasswordServer>
  Specifies the password that matches the super user password of the OS deployment server to which you link the web interface extension.

The computer from which you create the WinPE 3.x deployment engine can be:

- A local OS deployment server installed on a Windows operating system. This is the recommended option.
- Any computer with a Windows operating system.

From version 7.1.1.3 of the product, several WinPE 3.x deployment engines can coexist on any OS deployment server.

1. Depending on what you are doing, you can create the WinPE 3.x deployment engine from:
   - The **Deployment engine** page:
     a. Go to **Server** > **Advanced features** > **Deployment engines**.
     b. Click **New deployment engine**.
     c. Comlete the wizard.
   - The **Welcome** page:
     a. Select **Make one now** from the **For Windows scripted installation** or from the **For Windows clone installation** section.
     b. Click **Next**.
   - The **System profiles** page, if you do not already have a WinPE 3.x deployment engine:
     – If you run an unattended setup:
       a. Go to **Server** > **OS deployment** > **System profiles**.
       b. From the contextual menu, select **Add a new profile**.
       c. Select **Unattended setup (scripted install)**.
       d. Select one of the Windows operating systems as the type of system profile to create and click **Next**.
       e. The wizard displays a warning message informing you that it did not find a WinPE 3.x deployment engine. Click **Next** to create one.
     – If you run the capture of a cloned system profile:
       a. Go to **Server** > **OS deployment** > **System profiles**.
       b. From the contextual menu select **Add a new profile**.
       c. Select **Cloning from a reference machine**.
       d. Enter the IP address of the target that you want to clone. Ensure that the reference target is ready to boot into the OS deployment server and that it is shut down.

   e. The wizard displays a warning message informing you that it did not find a WinPE 3.x deployment engine. Click **Next** to create one.

2. Specify the address of the computer on which you installed Windows AIK for Windows 7 in English and the web interface extension and click **Next**.

You can now see the resulting WinPE 3.x deployment engines, one 32-bit WinPE 3.x deployment engine, and 64-bit WinPE 3.x deployment engine under **Server** > **Advanced features** > **Deployment engines**.

You can now indicate matching target models for your WinPE 3.x deployment engine and bind drivers to it.

After you create the WinPE 3.x deployment engines, you can create and deploy Windows system profiles.

Note: During the deployment, do not edit the WinPE 3.x deployment engine that you are using.

Creating aWinPE 4.x deployment engine

To create or deploy Windows 8 and Windows Server 2012 profiles, you must have created a WinPE 4.x deployment engine.

Ensure that the computer from which you create the WinPE 4.x deployment engine satisfies these conditions:

- Runs a Windows operating system
- Has Windows Assessment and Deployment Kit (ADK) for Windows 8 in English installed. Windows Assessment and Deployment Kit (ADK) for Windows 8 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34 &displaylang=en.
- Runs the appropriate web interface extension (rbagent). If the Windows operating system is 64-bit, stop the 32-bit web interface extension and start the 64-bit web interface extension as follows:

  ```
  C:\TPMfOS Files\global\http\rbagent64.exe -d -v 4 -s <IPServer>:<PasswordServer>
  ```

  where:

  <IPServer>
  Specifies the IP address of the OS deployment server

  <PasswordServer>
  Specifies the password that matches the super user password of the OS deployment server to which you link the web interface extension.

The computer from which you create the WinPE 4.x deployment engine can be:

- A local OS deployment server installed on a Windows operating system. This is the recommended option.
- Any computer with a Windows operating system.

Note: For non-deployment tasks that use a WinPE deployment engine, such as destroy hard disk content in kernel-free and Windows OS capture, the deployment engine used is the one selected by matching model pattern rules. If both WinPE 3.x and WinPE 4.x match the model pattern of the target machine, WinPE 4.x is used. If the target does not support WinPE 4.x (based on Windows 8 and the hardware requirements), to run one of these tasks ensure that model patterns for WinPE 4.x do not match the target model. For WinPE 4.x hardware requirement, see http://msdn.microsoft.com/en-us/library/windows/hardware/hh975398.aspx.

Several WinPE 4.x deployment engines can coexist on any OS deployment server.

1. Depending on what you are doing, you can create the WinPE 4.x deployment engine from:
   - The **Deployment engine** page:
     a. Go to **Server** > **Advanced features** > **Deployment engines**.
     b. Click **New deployment engine**.
     c. Complete the wizard.

        Note: In the "New deployment engine" wizard, if you select **Select the best available toolkit (Auto)** and both WinPE 3.x and WinPE 4.x are installed on the machine running the web interface extension, the system chooses WinPE 4.x.
   - The **Welcome** page:
     a. Select **Make one now** from the **For Windows scripted installation** or from the **For Windows clone installation** section.
     b. Click **Next**.
   - The **System profiles** page, if you do not already have a WinPE 4.x deployment engine:
     - If you run an unattended setup:
       a. Go to **Server** > **OS deployment** > **System profiles**.
       b. From the contextual menu, select **Add a new profile**.
       c. Select **Unattended setup (scripted install)**.
       d. Select one of the Windows operating systems as the type of system profile to create and click **Next**.
       e. The wizard displays a warning message informing you that it did not find a WinPE 4.x deployment engine. Click **Next** to create one.
     - If you run the capture of a cloned system profile:
       a. Go to **Server** > **OS deployment** > **System profiles**.
       b. From the contextual menu select **Add a new profile**.
       c. Select **Cloning from a reference machine**.
       d. Enter the IP address of the target that you want to clone. Ensure that the reference target is ready to boot into the OS deployment server and that it is shut down.
       e. The wizard displays a warning message informing you that it did not find a WinPE 4.x deployment engine. Click **Next** to create one.
2. Specify the address of the computer on which you installed Windows ADK for Windows 8 in English and the web interface extension and click **Next**.

You can now see the resulting WinPE 4.x deployment engines, one 32-bit WinPE 4.x deployment engine, and one 64-bit WinPE 4.x deployment engine under **Server** > **Advanced features** > **Deployment engines**.

If WinPE 3.x deployment engines are already available on the OS deployment server, WinPE 4.x deployment engines are not bound to any model pattern automatically. To add the matching model patterns see “Adding matching target models to a WinPE deployment engine” on page 26.

You can now indicate matching target models for your WinPE 4.x deployment engine and bind drivers to it.

After you created the WinPE 4.x deployment engines, you can create and deploy Windows system profiles.

Note: During the deployment, do not edit the WinPE 4.x deployment engine that you are using.

Editing the information of a WinPE deployment engine

You can edit the description and the comment attached to a WinPE deployment engine.

To edit the description and comment of a deployment engine:

1. Go to **Server** > **Advanced features** > **Deployment engines**.
2. To view the details of the deployment engine, perform one of the following options:
   - Double-click a deployment engine.
   - Select a deployment engine, and then select **View engine details** in the contextual menu.
3. Click **Edit** above the section **Deployment engine information**.
4. Update the description and the comment to identify more easily how this WinPE deployment engine is to be used.
5. Click **OK** to save your changes and return to the **Engine details** page.

If you intend to use this deployment engine to deploy IBM servers, you might want to call your WinPE deployment engine `WinPE3 for IBM servers 32-bit`. The comment can include the server models that this WinPE deployment engine is planned to be compatible with.

If you updated the description of your WinPE deployment engine, you probably have more than one deployment engine per architecture. In this case, provide matching target models for your deployment engines.

Upgrading WinPE deployment engines and hardware environments

When upgrading to Fix Pack 12, you must upgrade the WinPE deployment engines and the WinPE 2.x or 3.x hardware environments.

If old WinPE (deployment engines and hardware environments) engines are detected, a warning message and an icon is displayed in the web interface near each object.

If the Windows Automated Installation Kit (AIK) is installed on the same workstation where the OS deployment server is installed, when the OS deployment server starts up, all WinPE deployment engines (not hardware environments, only deployment engines) are automatically upgraded.

A wizard and a command line with rbagent are available to update the WinPE engines (deployment engines and hardware environments).

The following command line with rbagent is available to update the WinPE engines:

```
rad-makewpe update=EngineItemID
rad-mkhwutils update=SoftItemID
```

For the *EngineItemID*, see the EngineItem table. For the *SoftItemID*, see the SoftwareItem table.

Adding matching target models to a WinPE deployment engine

If you have several WinPE deployment engines for the same architecture, you must specify with which targets a given WinPE deployment engine must be used.

If you have only one WinPE deployment engine per computer architecture, there is no reason to modify the model patterns. Use the default * pattern, to match any target known to the OS deployment server.

To add model patterns associated with a deployment engine:

1. Go to **Server** > **Advanced features** > **Deployment engines**.
2. To view the details of the deployment engine, perform one of the following options:
   - Double-click a deployment engine.
   - Select a deployment engine, and then select **View engine details** in the contextual menu.
3. In the **Matching models** section, click **Add a new model pattern**.
4. Enter the pattern and click **OK** to save your new pattern. The **?** character is used as a wildcard replacing one character.

   Note: If WinPE 3.x deployment engines are already available on the OS deployment server, WinPE 4.x deployment engines are not bound to any model pattern automatically.

When deploying a target, if there are several WinPE deployment engines available, a search is performed in the list of model patterns for all WinPE deployment engines available. The WinPE deployment engine selected has the most restrictive pattern matching the target model being deployed.

If there is no matching pattern, deployment cannot proceed.

Note: In a multiple server architecture, a WinPE deployment engine that is not fully replicated from a parent server is not yet available on the child server.

Consider that you have two WinPE deployment engines, `WinPEa` and `WinPEb`. `WinPEa` has the following patterns: `IBM Server *`, and `lenovo *`, while `WinPEb` has `lenovo m/55 *`, `lenovo T*`, and `*`.

A target with model `lenovo T61` is deployed with `WinPEb` because its model matches the `lenovo T*` pattern, because it is more restrictive than `lenovo *`.

A target with model `lenovo ThinkCenter A58` is deployed with `WinPEa` because its model matches the `lenovo *` pattern, because it is more restrictive than the generic `*` pattern.

A target with model `HP Server` is deployed with `WinPEb` because its model matches only the `*` pattern.

You can check which WinPE deployment engine is used with a given target by looking at the **Windows specific info** section in **Server** > **OS deployment** > **Target Monitor** > **Target details**. If you are dissatisfied with the selected WinPE deployment engine, you must adapt the target models for your WinPE deployment engines.

Binding drivers to a WinPE deployment engine

When WinPE does not contain the drivers that you need for a specific target, you must bind drivers to the WinPE deployment engine to deploy the target.

Your WinPE deployment engine contains built-in drivers. Use them first.

If you encounter problems with the built-in drivers, if some drivers are not bound, or if some drivers are missing, bind other drivers to your WinPE deployment engine in one of the following ways:

- Static driver injection
- Dynamic driver injection

Note: For drivers that are not PCI-based (no PCI strings exist in the `.inf` files), you must manually bind them using static driver injection.

In the static driver injection process, you can only bind drivers, to your WinPE deployment engine, that are driver software modules in your OS deployment server. You must therefore create driver software modules from the drivers that you want to bind to your WinPE deployment engine.

The product helps you select appropriate drivers for particular target models. It helps you to predict potential problems and to solve them. It does not guarantee that a specific WinPE deployment engine, with bound drivers, works with a given target.

The information used by the OS deployment server to predict the compatibility of a driver with a target model is taken from the content provided by the vendor in its driver. The OS deployment server cannot verify the accuracy of this information.

The dynamic driver injection process occurs at run time and depends on the model and PCI devices. The following is a high-level view of the dynamic driver injection process:

1. WinPE3 is started.
2. The web interface extension is started in WinPE3.
3. The web interface extension determines the list of drivers.
4. The web interface extension detects the hardware on which it is running.
5. The web interface extension injects only the drivers specifically bound.

Note: When using automatic driver binding, if the hard disk comes before the network in the boot order, if the machine automatically reboots, you must force the boot on the network.

1. Check the compatibility of your WinPE deployment engine.
   a. Go to **Server** > **Advanced features** > **Deployment engine**.
   b. To view the details of the deployment engine, choose one of the following options:
      - Double-click a deployment engine.

- Select a deployment engine, and then select **View engine details** in the contextual menu.

c. Go to the section **Network and mass storage drivers**. A check is performed while the page is loading. This can take a few minutes. By default, checks are performed only on network and disk drivers.

If drivers are missing, or are not bound, or if several drivers are bound for the same device, the following information is provided:

❌ Indicates a missing critical driver, or a critical driver of the wrong architecture.

✖ Indicates that a missing non-critical driver, or a non-critical driver of the wrong architecture.

⚠ Indicates that a required driver is present on the OS deployment server, but that it is not bound.

ℹ Indicates that there are several drivers bound for the same device, or that there is a binding with a driver that is not known as compatible.

You can expand the item to get more information.

- For drivers missing on the OS deployment server, you can discover where to look for it, including, if available, a download link and the exact directory within the downloaded archive where the driver can be found.
- When drivers are present on the OS deployment server, you can discover which driver to bind, in order of preference. If multiple drivers are known to possibly work for a device, the best choice is listed first. The choice is explained in the advice text, which first recommends the use of *device-specific drivers*, that is, drivers that have been specifically designed for the given hardware device. Then *compatible device drivers*, that match the device family, are recommended, even if they are not an exact rebranded variant (for example, as second choice, an Adaptec driver of the same family as an IBM® ServerRaid adapter, if it is based on the same chipset). Finally, as third choice, *generic drivers*, for example, Microsoft generic AHCI driver for any AHCI controller, are recommended.

If no error is found, you do not need to modify the bindings.

2. Modify the driver bindings of the WinPE deployment engine. You can do this in the following ways:
   - Use a wizard.
     a. Click **Fix Drivers**.
     b. Follow the instructions in the wizard. After having selected a target model, you must select one of these options:

Automatically fix issues that can be fixed for this model.
Fixes all issues that can be automatically fixed. Such issues include, for example, a missing binding to an existing driver, multiple bindings for a device, or removing a driver tagged for another operating system.

Manually fix issues for this model.
Presents you with each issue in turn. Ways to solve the issue, when available, are proposed.

Automatically bind drivers for this model.
Erases every existing binding. New bindings are then automatically added.

Copy driver bindings for this model from a similar engine.
Copies all the bindings from a selected source engine to the current engine.

Reset all drivers bindings for this model.
Erases all the driver bindings, and does not create any new binding.

- Edit the bindings manually, using the driver binding grid.
  a. Click **Edit engine's driver bindings** on the **Engine details** page.

     A grid is loaded.

     Columns represent target models known to the OS deployment server and matching the patterns provided for the WinPE deployment engine. They can be expanded to view their network and mass storage devices, if a PCI inventory has been performed.

     The first line represents the WinPE deployment engine. Other lines represent software module folders in the OS deployment server. They can be expanded to view individual drivers. If a driver can be used only for 32-bit or 64-bit machines, a superscript *x86* or *x86-64* mark is written next to the driver name. If you do not find the drivers that you need in the list provided, create software modules for your drivers.
  b. Optional. To obtain a summary of the errors and warnings, click the link above the grid. This helps you locate the problematic areas in the driver grid.
  c. Expand the columns of problematic target models to view the individual network and mass storage devices.
  d. Expand software module folders containing drivers to view the individual drivers.

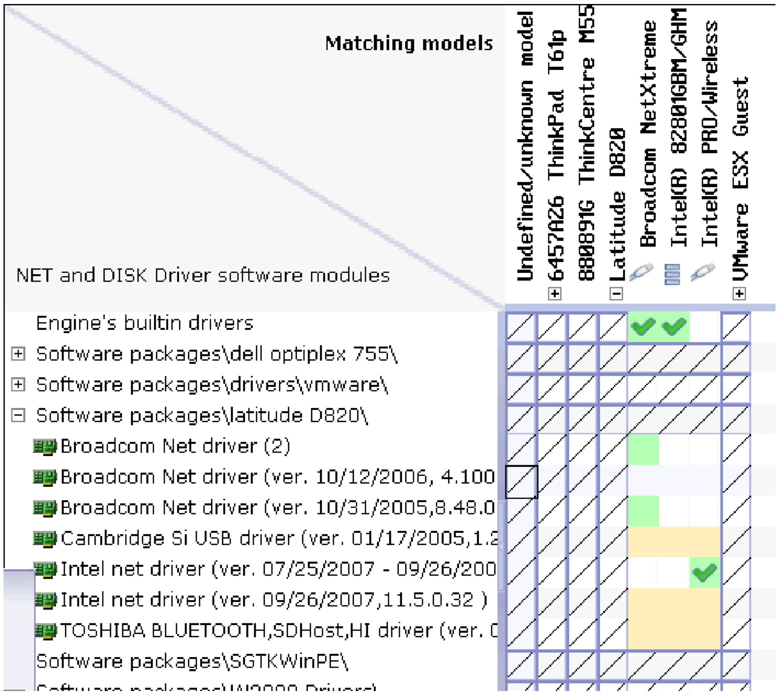

Matching models
NET and DISK Driver software modules
Undefined/unknown model
⊞ 6457A26 ThinkPad T61p
8808916 ThinkCentre M55
⊟ Latitude D820
Broadcom NetXtreme
Intel(R) 82801GBM/GHM
Intel(R) PRO/Wireless
⊞ VMware ESX Guest
Engine's builtin drivers
⊞ Software packages\dell optiplex 755\
⊞ Software packages\drivers\vmware\
⊟ Software packages\latitude D820\
Broadcom Net driver (2)
Broadcom Net driver (ver. 10/12/2006, 4.100
Broadcom Net driver (ver. 10/31/2005,8.48.0
Cambridge Si USB driver (ver. 01/17/2005,1.2
Intel net driver (ver. 07/25/2007 - 09/26/200
Intel net driver (ver. 09/26/2007,11.5.0.32)
TOSHIBA BLUETOOTH,SDHost,HI driver (ver. 0
Software packages\SGTKWinPE\

Figure 2. Driver binding grid

A cell with a green background indicates that driver information corresponds to the device. The quality of the drivers that can be selected is illustrated by the intensity of the green background: ■ ■ ■ the best drivers are in intense green, the family drivers are in standard green, and the generic drivers are in pale green.

A cell with an orange background indicates either that the driver is not a PCI driver, or that there is no compatibility information available for the driver.

A cell with a green check mark ✓ indicates that the driver is bound to the WinPE deployment engine for use with the specific target model and device.

e. Click a green or orange background cell to add or remove bindings.

   You cannot bind or unbind drivers from the WinPE deployment engine itself, because they are built-in drivers.

   You must have only one check mark per column, indicating that you have only one driver for each device.

f. When you have finished modifying the bindings, click **Save**.

g. To return to the **Image details** page and click **Back**.

   Potential problems with the image are recomputed, so you can check if your modifications have solved the detected problems.

When you have solved all the driver binding issues, you can deploy target models that match your WinPE deployment engine.

System profiles for Windows operating systems

A system profile is the partition layout and list of files to deploy an operating system, either by unattended setup or by cloning, from a reference target or from a reference image file.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.

- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.
- The exFAT filesystem is not supported.
- Before deploying a system profile to a target ensure that the root partition is C.
- To successfully implement the user category options, ensure that the system profile disk layout label does not contain a semicolon ":".
- It is possible to deploy a system profile captured from a BIOS machine to a UEFI machine. During this operation, ESP and MSR are added on the fly and patched accordingly.

BitLocker compatibility

Tivoli Provisioning Manager for Images is compatible with Microsoft BitLocker Drive Encryption (BitLocker), which is available with some Windows operating systems. BitLocker is a security tool which protects data by encrypting it, rendering the content of a hard disk unreadable if stolen.

Windows 7

BitLocker on Windows 7 Ultimate and Enterprise operating systems

To operate on Windows 7 operating systems, BitLocker requires a minimum of 300 MB of unallocated space (not part of a partition) on the target disk.

Vista

BitLocker on Windows Vista operating systems

To operate on Windows Vista operating systems, BitLocker requires at least two partitions:

- a boot partition containing the BitLocker tool and which must have a size of at least 1.5 GB
- an operating system partition which can be encrypted

Tivoli Provisioning Manager for Images can make a deployed target ready for BitLocker by creating the appropriate partition scheme during the deployment.

When you create a system profile for Windows Vista/2008/7/2012, the Profile Wizard asks you whether you want to make your profile ready for BitLocker. In case of a positive answer, the wizard asks you the relevant questions to set up the partition scheme.

Note: When you run Microsoft System Preparation Tool (Sysprep) on a BitLocker ready target, which is necessary for cloning, Sysprep deletes some vital information about the boot and the operating system partitions. It results in a reference target which cannot boot anymore. During the cloned profile creation process, Tivoli Provisioning Manager for Images can partially repair the reference target to make it boot again. However, some manual operations with Microsoft tools remain necessary to make it BitLocker ready again.

To create a cloning profile from a BitLocker ready reference target and have this reference target operational and BitLocker ready again:

1. Make sure that the disk is not encrypted.
2. Run Sysprep on the reference target

3. In the Profile Wizard, select the option to repair the reference target to enable the target to boot again.
4. Manually modify the boot and operating system partitions with Microsoft tools to make the partition scheme BitLocker ready again.

Alternatively, if you do not want to perform manual operations to make your reference target BitLocker ready again, you can

1. Make sure that the disk is not encrypted.
2. Run Sysprep on the reference target
3. Create the cloned system profile
4. Deploy the reference target with the newly created cloned profile which is BitLocker ready

Enabling the BitLocker feature and creating the partition after the deployment for Windows 7 and Windows 2008 R2

This topic describes how to create the Bitlocker partition using Microsoft Bitlocker on Windows 7 and Windows 2008 R2 platforms.

Scenario

For Windows 7 and Windows 2008 R2 platforms, the solution is to run the Bitlocker command to create the partition using the **bdehdcfg** command. It is recommended to run the **bdehdcfg** command after the deployment is completed. This can be done, for example, by using the runonce key in the Windows registry. To enable the BitLocker feature and have the partition created after the deployment, perform the following procedure.

Procedure

1. Create a software module that runs the command:

   ```
   cmd.exe /C start c:\Windows\system32\Dism /online /enable-feature
    /featurename:BitLocker /NoRestart /quiet
   ```

   This enables the BitLocker feature needed to run the **bdehdcfg** command.
2. Create a .bat script (named for example bitlockerpart.bat) including the following command:

   ```
   cmd.exe /C start c:\windows\system32\bdehdcfg
   -target c: shrink -size 300 -quiet
   ```

   and create a software module that copies the .bat file onto a directory on the target disk, for example on the C: drive.
3. Create a software module which adds the .bat execution into the RunOnce registries key, that is a software module running the following command:

   ```
   cmd.exe /C start c:\windows\system32\reg add HKEY_LOCAL_Machine\
   Software\Microsoft\Windows\CurrentVersion\RunOnce /v bitlocker
   /t REG_SZ /d "c:\bitlockerpart.bat"
   ```

   The c:\bitlockerpart.bat file is run automatically at the first user logon.

All these software modules can be run during the stage when the operating system is installed. In this way the BitLocker feature is enabled and the partition is created at the first user logon after the deployment finishes successfully.

Creating system profiles

There are distinct types of system profiles. The profile wizard guides you through the creation of system profiles for each type.

Creating an unattended setup system profile for Windows operating systems

You can install operating systems using standard installation processes in unattended mode. Unattended setup simplifies the task of preparing computers for the native mode of operation of disk cloning.

- To create a Windows system profile you must have a WinPE 3.x or WinPE 4.x deployment engine on your OS deployment server. If you do not have one yet, you can create one with the profile wizard, provided you have installed Windows AIK for Windows 7 in English or Windows ADK for Windows 8 in English on the computer on which you create the WinPE 3.x or WinPE 4.x deployment engine.

Vista 2008 Windows 7

- To create an unattended Windows Vista/2008/7 setup system profile, you must use a computer running the web interface extension, where you have installed Windows AIK for Windows 7 in English, under Windows XP, Windows 2003, Windows Vista, Windows 2008, or Windows 7. To create an unattended Windows 8 or Windows 2012 setup system profile, you must use a computer running the web interface extension, where you have installed Windows ADK for Windows 8 in English under Windows Vista, Windows 2008, Windows 7, Windows 8 and Windows 2012. You cannot run this operation on a Windows 2000 or Linux operating system. The web interface extension must be started with local administrator privileges.

  Note: For Windows Vista, Windows 2008, and Windows 7 unattended system profiles:

- During the various **autoadminlogon**, the user accessing the machine is the one specified as administrator account at operating system configuration level. If the administrator account is the same as the built-in administrator at operating system configuration level, the built-in administrator account is enabled.

  If the administrator account is not specified or if the string **Administrator** is specified, the **autoadminlogon** is executed with the built-in administrator account which is enabled at the end of the deployment.

  For example, if your system is in Spanish and you specify the administrator account in one of the three different ways:

  – administrator account = noone

  – administrator account = Administrador (in Spanish)

  – administrator account = Administrator (in English)

  then you can execute the **autoadminlogon** with the built-in administrator account and the built-in administrator is enabled.

- Creating an unattended Windows Vista/2008/7/2012/8 installation profile with multiple CDs is not supported. You are required to use a single DVD.
- You can prepare your profile to be ready for Microsoft BitLocker Drive Encryption (BitLocker).

  Windows 7 Modify the partition scheme to ensure that there is at least 300 MB of unallocated space on the disk.

  Vista You must have at least two partitions:

- A partition of at least 1.5 GB is necessary to hold BitLocker and to serve as a boot partition
- A second partition holds the operating system

Depending on the number of partitions already created, the Profile Wizard offers to reserve one of the existing partitions for BitLocker, or to create a new one.

2003 The Windows 2003 R2 operating system is distributed on two CDs. To create a fully deployable unattended system profile of Windows 2003 R2, you must:

1. Create a system profile using the first CD only, following the steps in the wizard;
2. Create a software module with the content of the second CD (see "Creating a software module for unattended deployment of Windows 2003 R2 operating system" on page 70);
3. Bind this software module (with an automatic binding rule) to the system profile you just created.

To create a new system profile:

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select your operating system from the list and click **Next**.
5. Follow the instruction of the profile wizard. Vista 2008 Windows 7 If you have a volume license, select **Volume licensing, no product key required** on the product key screen of the wizard.

   Note: Tivoli Provisioning Manager for Images supports the RETAIL KEY only. Any volume license key, either MAK or KMS, entered into a Windows profile will not work. If you only have a volume license key, select **volume license** on the product key screen of the wizard and create a software module that installs the volume license key through the command **`slmgr.vbs`**.

When your first unattended installation profile is created, you can use it to deploy targets. Then you can create a cloning-mode system profile, because unattended installation profiles have a longer deployment time than cloning-mode system profiles. You can use your unattended installation profile to prepare the computer that you refer to when creating your first cloning-mode system profile.

Creating a cloning mode system profile for Windows operating system

To obtain a cloning-mode system profile from a reference target you must first prepare the reference target.

To clone a Windows operating system, your reference target must have at least 1 GB RAM.

Note: For Windows Vista, Windows 2008, and Windows 7 clone system profiles:

During the various **`autoadminlogon`**, the user accessing the machine is the one specified as administrator account at operating system configuration level. If the administrator account is the same as the built-in administrator at operating system configuration level, the built-in administrator account is enabled. If the administrator account is not specified, the **autoadminlogon** is executed with the built-in administrator account which is enabled at the end of the deployment.

To create a Windows system profile you must have a WinPE 3.x or WinPE 4.x deployment engine on your OS deployment server. If you do not have one yet, you can create one with the profile wizard, provided you have installed Windows AIK for Windows 7 in English or Windows ADK for Windows 8 in English on the computer on which you create the WinPE 3.x or WinPE 4.x deployment engine.

For the actual driver injection, you must use a computer running the web interface extension, where you have installed Windows AIK for Windows 7 in English, under Windows XP, Windows 2003, Windows Vista, Windows 2008, or Windows 7. To create an unattended Windows 8 or Windows 2012 setup system profile, you must use a computer running the web interface extension, where you have installed Windows ADK for Windows 8 in English under Windows Vista, Windows 2008, Windows 7, Windows 8 and Windows 2012. You cannot run this operation on a Windows 2000 or Linux operating system. The web interface extension must be started with local administrator privileges.

Note: If both WinPE 3.x and WinPE 4.x are available, WinPE 4.x is used. If target does not support WinPE 4.x, to run this task ensure that model patterns for WinPE 4.x do not match the target model. WinPE 4.x is used only to provision Windows 8 and Windows 2012. For information about Windows 8, Windows 2012 and WinPE 4.x, see http://msdn.microsoft.com/en-us/library/windows/hardware/hh975398.aspx.

Preparing the reference target:

To create a cloning-mode system profile, you must first create the reference OS configuration, the *system profile*, that you want to deploy.

You must perform this task on the reference target and not on the OS deployment server.

The OS deployment server does not perform cleanup on the reference target. You must delete useless files and services before creating a new image:
- Delete the temporary internet cache.
- Delete your temporary directories and files.
- Disconnect your network drives and remote printers.
- Empty the recycle bin.
- Delete partitions that use a file system not supported by the product, or reformat the partitions.

Additional preparation instructions:

This section describes how to extract and decrypt all files on the disk to include them in the disk image.

When installing Windows 2000 or Windows XP on an NTFS partition, some files are compressed by default. In particular, the `dllcache` hidden folder and the `explorer.exe` file are usually compressed. This might prevent them to be backed up or imaged correctly. To extract and decrypt all files on your disk, perform these steps:

1. Open a command window.

2. Type the following commands:

```
cd \
COMPACT /u /a /s /i
CIPHER /d /a /i /f /h /s:\
exit
```

The COMPACT command will automatically and recursively extract all files on the disk, while the CIPHER command will automatically and recursively decrypt all these files.

Running Sysprep:

Before you can create a cloning-mode system profile for Windows operating system, you must run Microsoft System Preparation Tool (Sysprep). Where to find Sysprep and how to use it varies slightly depending on the Windows version.

Running Sysprep on Windows Vista/2008/7/2012 operating systems:

Before cloning your Windows Vista/2008/7/2012 image, run Sysprep to prepare your system for cloning. Tivoli Provisioning Manager for Images works with Sysprep to automate the post-cloning reconfiguration.

Sysprep cannot be used on targets that are part of a domain. The system profile image must be made on a target hat does not belong to a domain. Even if your operating system was part of the domain before you launched Sysprep, Sysprep removes it from the domain. Later, you can automatically join a domain during the deployment process.

Before running Sysprep, you must configure your target to use DHCP. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Then click on **Common networking info**. If your target uses a static IP address, you have a high risk IP conflicts when the target boots for the first time and it has not yet applied all Sysprep settings.

With Windows Vista/2008/7/2012, you can run Microsoft System Preparation Tool (Sysprep) on the operating system only three times. After that, the Sysprep tool refuses to start, therefore always start from your original reference image. To work around this issue, you can also use a virtual machine.

Sysprep is available on every installed Windows Vista/2008/7/2012 operating system. The Sysprep executable file is archived in `c:\windows\system32\sysprep\sysprep.exe`.

To start the Sysprep process, follow these instructions:

1. Log on as a user with administrator privileges.
2. Close any open applications and type the run command in the Windows Vista/2008/7/2012 **Start Search** command prompt.
3. When the run command prompt opens, browse to the Sysprep executable file and click **OK**. A System Preparation Tool page opens.
4. From the System Cleanup action menu, select **Enter System Out-of-Box Experience (OOBE)**.
5. Select the **Generalize** check box.
6. From the **Shutdown Options menu**, select **Shutdown**.
7. Click **OK**. After a few seconds, your system shuts down automatically.

Alternatively, you can specify these options when launching Sysprep from the command line prompt by running the command: `c:\windows\system32\sysprep\sysprep.exe /oobe /generalize /shutdown`.

Note:

- Sysprep can also be used in audit mode. In audit mode, when the user first boots the deployed machine, the boot process does an Out-Of-Box Experience (OOBE) stage which finalizes the OS configuration taking connected peripherals into account. This OOBE stage takes about 10 minutes. If Sysprep is used in OOBE mode, this stage is performed during deployment without significantly increasing the deployment time.
- It is possible to have a partition dedicated to Microsoft BitLocker Drive Encryption (BitLocker).
  - If the reference computer you are cloning is BitLocker ready, running Sysprep prevents it to boot anymore.The product can correct this error and allow the computer to boot again by assigning the operating system partition as boot partition. However, if you want to use BitLocker on the reference target afterward, you must manually change the boot partition back to the BitLocker partition.The product properly configures boot and root partitions on deployed computers. Thus, computers deployed with an image cloned from a BitLocker ready computer are perfectly bootable and BitLocker ready.
  - If the reference computer is not BitLocker ready, running Sysprep does not raise any difficulty. To make the cloned target BitLocker ready, modify the partition layout of the system profile. For Windows 7 operating systems, you need at least 300 MB of unallocated disk space. For Windows Vista operating systems to assign or create a BitLocker partition of at least 1.5 GB.

Running Sysprep on Windows XP and Windows 2003 operating systems:

Before cloning your Windows image, run Sysprep to prepare your system to be cloned. Tivoli Provisioning Manager for Images works with Sysprep to automate the post-cloning reconfiguration.

Sysprep cannot be used on targets that are part of a domain. The system profile image must be made on a target hat does not belong to a domain. Even if your operating system was part of the domain before you launched Sysprep, Sysprep removes it from the domain. Later, you can automatically join a domain during the deployment process.

Before running Sysprep, you must configure your target to use DHCP. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Then click on **Common networking info**. If your target uses a static IP address, you have a high risk IP conflicts when the target boots for the first time and it has not yet applied all Sysprep settings.

Depending on how Windows was installed, you might never have logged on as an administrator. If this is the case, log out and log in again as an administrator to ensure that the administrator profile is properly created. Otherwise, you might not be able to create system snapshots affecting the administrator settings.

Sysprep for Windows XP is included on the Windows XP Professional CD, and archived in the file `\Support\Tools\Deploy.cab`.

To run Sysprep:

1. Copy all the Sysprep executable files into a folder named `c:\sysprep`.
2. Close all your applications.
3. Run the command `c:\sysprep\sysprep.exe -mini -forceshutdown -reseal` from the **Start > Run** menu.

   Alternatively, you can start Sysprep with a graphical user interface by double-clicking on its icon

   a. Make sure that **Mini Setup** is checked
   b. Click **Reseal**.

   Your system shuts down automatically after a few seconds.

Running Sysprep on Windows 2000 operating system:

Before cloning your Windows 2000 image, run Microsoft system preparation tool (Sysprep) to prepare your system to be cloned. Tivoli Provisioning Manager for Images works with Sysprep to automate the post-cloning reconfiguration.

Sysprep cannot be used on targets that are part of a domain. The system profile image must be made on a target hat does not belong to a domain. Even if your operating system was part of the domain before you launched Sysprep, Sysprep removes it from the domain. Later, you can automatically join a domain during the deployment process.

Before running Sysprep, you must configure your target to use DHCP. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Then click on **Common networking info**. If your target uses a static IP address, you have a high risk IP conflicts when the target boots for the first time and it has not yet applied all Sysprep settings.

Sysprep for Windows 2000 is included in Windows 2000 Resource Kit, and is also available on the Microsoft Web site.

1. Copy all the Sysprep executable files into a folder named `c:\sysprep`.
2. Close all your applications.
3. Run the command `sysprep.exe` from the **Start > Run** menu. Your system shuts down automatically after a few seconds (if it does not, wait a minute or so and then turn it off).

Cloning the reference computer:

After you have prepared your reference computer, you can create your system profile from the profile wizard in the web interface.

- The reference target must be ready to PXE-boot into the OS deployment server.
- Shut down the the reference target before starting the cloning process. If it is not shut down, ensure that the web interface extension is not running.

1. Go to **Server** > **OS deployment** > **System Profiles**.
2. Click **New profile**.
3. Select **Cloning from a reference machine** and click **Next**.
4. Complete the profile wizard. If you have a volume license, select **Volume licensing, no product key required** on the product key screen of the wizard.

   Note: Tivoli Provisioning Manager for Images supports the RETAIL KEY only. Any volume license key, either MAK or KMS, entered into a Windows profile will not work. If you only have a volume license key, select **volume license** on the product key screen of the wizard and create a software module that installs the volume license key through the command **`slmgr.vbs`**.

The **Volume licensing, no product key required** option does not work for all versions of Windows operating systems. If you are asked for a deployment key during deployment, go to **Server** > **System profiles** > **Profile details** > **OS configuration details**, click on the **Windows** tab and set **Volume Licensing** to **No**. You must then populate **Product key**. Your cloning system profile should now deploy without userinteraction.

Creating a system snapshot:
You can clone a computer with a Windows operating system without running the Sysprep tool. Such a profile is called a system snapshot. You can create it from the web interface.

1. Go to **Server** > **OS deployment** > **System Profiles**.
2. Click **New profile**.
3. Enter the IP address of the target you want to clone and click **Next**.
4. The Profile Wizard detects the operating system. Click **Next**.
5. The Profile Wizard detects that the Windows operating system has not been prepared with Sysprep. Review the warnings carefully and, if you still intend to create a system snapshot, select **I understand these limitations but I want to proceed as is** and click **Next**.
6. Follow the instructions of the wizard.

System snapshot:

You can create Windows cloning profiles without using Sysprep to prepare your reference target. Such a profile is called a system snapshot.

The purpose of creating a system snapshot is to keep a copy of a golden parent reference target before it is altered by Sysprep, thus enabling you to restore your golden parent exactly as it was before the Sysprep tool was used.

Note:

1. The product had not been designed as a backup product.
   - Do not create and restore Windows system snapshots as a backup method.
   - Do not create more than a few Windows system snapshots on any OS deployment server.
2. You cannot deploy a Windows system snapshot, you can only restore it exactly as it was created.
   - It is not possible to customize system snapshots.
   - Profile restoration does not allow the installation of software modules, including driver packages.

Creating a system profile from a reference image

You can create a system profile using a WIM image.

- You can create system profiles from WIM image for Windows Vista/2008/7/2012, Windows XP and Windows 2003 operating systems.
- The WIM image present on Microsoft DVD is not a complete operating system image; therefore, it cannot be imported into Tivoli Provisioning Manager for Images to be deployed. To create a system profile from Microsoftlist DVD, use the option `Unattended setup (scripted install)`.

- The WIM image must contain only one partition. If you have two partitions in your WIM image, for example, a boot partition and a separate root partition, deployment of the cloning WIM system profile fails.
- To create a Windows system profile you must have a WinPE 3.x or WinPE 4.x deployment engine on your OS deployment server. If you do not have one yet, you can create one with the profile wizard, provided you have installed Windows AIK for Windows 7 in English or Windows ADK for Windows 8 in English on the computer on which you create the WinPE 3.x or WinPE 4.x deployment engine.
- To create a cloning profile with a Windows WIM image, you must use a computer running the web interface extension, where you have installed Windows AIK for Windows 7 in English, under Windows XP, Windows 2003, Windows Vista, Windows 2008, or Windows 7. To create an unattended Windows 8 or Windows 2012 setup system profile, you must use a computer running the web interface extension, where you have installed Windows ADK for Windows 8 in English under Windows Vista, Windows 2008, Windows 7, Windows 8 and Windows 2012. You cannot run this operation on a Windows 2000 or Linux operating system. The web interface extension must be started with local administrator privileges.
- Creating a cloning profile from a Windows WIM image stored on multiple CDs is not supported. You are required to use a single DVD.
- You can prepare your profile to be ready for Microsoft BitLocker Drive Encryption (BitLocker).

  Windows 7 Modify the partition scheme to ensure that there is at least 300 MB of unallocated space on the disk.

  Vista You must have at least two partitions:

  – A partition of at least 1.5 GB is necessary to hold BitLocker and to serve as a boot partition

  – A second partition holds the operating system

  Depending on the number of partitions already created, the Profile Wizard offers to reserve one of the existing partitions for BitLocker, or to create a new one.

To create a system profile from a reference image, you must follow these steps:

1. Go to **Server** > **OS deployment** > **System Profiles**.
2. Click **New Profile**. This opens a system profile wizard that guides you through the steps of creating a profile.
3. Select **Cloning from a reference image file** and click **Next**.
4. Select the corresponding image format and click **Next**.
5. Follow the instruction of the profile wizard. If you have a volume license, select **Volume licensing, no product key required** on the product key screen of the wizard.

   Note: Tivoli Provisioning Manager for Images supports the RETAIL KEY only. Any volume license key, either MAK or KMS, entered into a Windows profile will not work. If you only have a volume license key, select **volume license** on the product key screen of the wizard and create a software module that installs the volume license key through the command **`slmgr.vbs`**.

Creating a universal system profile for Windows operating systems

When creating a software module, do not enter a hardware model because a universal system profile must be deployable on several types of hardware. If you entered a model name in the Profile Wizard, you can delete it when you edit the first set of parameters of the Profile details.

To successfully deploy your universal system profile with another type of hard disk from your reference target (for example from a parallel hard disk to a SCSI or an AHCI disk), you must inject the drivers during deployment.

There are two different scenarios

- In an unattended setup system profile, the driver MUST be injected as a software module created by Tivoli Provisioning Manager for Images.
- In a cloning system profile, the driver might be injected as a software module created by Tivoli Provisioning Manager for Images. If this method fails, you can use the Microsoft "Sysprep" tool.

Here are the solutions for these two different scenarios.

Deploying an unattended setup:
When a driver needs to be installed during the early stages of Windows unattended setup, you must use TEXTMODE drivers. Perform these steps:

1. Ensure that the driver files are on your server. The file txtsetup.oem must be in the driver folder. This file is provided by the hardware vendor.
2. Create a software module, type driver. Typically, Tivoli Provisioning Manager for Images recognizes that this is a TEXTMODE driver and completes the fields automatically. The installation stage must be “When the OS is installed”.
3. Bind your software module to your target profile, or bind it automatically to the hardware.
4. You can now deploy your unattended profile with the software module.

The above solution might not work when deploying a cloned system profile because Windows setup does not use the same mechanisms as Windows Sysprep for handling mass storage drivers.

Deploying a cloning system profile:

Tip: The easiest and safest solution to deploy a system profile is to start from a computer which has similar hardware to the target system, in particular regarding mass storage drivers. This will save you time and make the process deployment easier to understand and follow.

If the driver injection using the mechanism of a software module created by Tivoli Provisioning Manager for Images fails, you can use the Microsoft Sysprep tool.

You must inject the driver into the parent system profile, by performing some extra steps on the source computer before running the Sysprep tool to reseal the system profile. Prepare the drivers that you want to inject into your clone system profile in a separate folder. Perform these steps:

1. Place all the driver files on the source computer into a `C:\drivers\MyDiskController` folder.
2. Create a "Sysprep.inf" file that you place in the `c:\Sysprep` folder with the correct settings.
3. Run Sysprep.
4. Capture the cloned system profile.

When deploying the system profile, the driver injected on the target system is automatically enabled when the system starts up.

The source computer uses an EIDE controller (any type).

The target system uses an IBM ServRAID 8i controller: 0x9005(VendorID) 0x0285(DeviceID)

Copy all driver files for this controller into `arcsas.sys` and `arcsas.inf`, including all the files referenced from this file.

In your `Sysprep.inf` file, copy the following section:

```
[Sysprepmassstorage]PCI\VEN_9005&DEV_0285&SUBSYS_02f21014="%SystemDrive%\
drivers\ServeRAID8i\arcsas.inf", "\", "IBM ServeRAID 8i Controller",
"\arcsas.sys"
```

Tip:

- All the PCI numbers can be found in the web interface, in the hardware inventory tab of the target.
- To determine the appropriate driver, check the **PCI_VENxxxx&DEV_xxxx** string in the driver inf file and match it with the data reported on target hardware inventory, as reported by TPM for OS deployment. The SUBSYS_yyyyzzzz must also match the **SubVendorID (yyyy)** and **SubDeviceID (zzzz)**.
- The PCI\... key that you add to your Sysprep.inf file must be an exact copy of the one used in the driver .inf file.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles

To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Updating a system profile with a Language Pack or a HotFix

Vista 2008 Windows 7

Windows Vista/2008/7/2012 system profiles can be updated to include a language pack or a HotFix.

To update a system profile to include either a Language Pack, or a HotFix, or both, you need an available target on which the profile will be updated.

Note:

- To update an unattended setup profile or a WIM cloning profile, you must use a computer running the web interface extension, where you have installed Windows AIK for Windows 7 in English, under Windows XP, Windows 2003, Windows Vista, Windows 2008, or Windows 7. To create an unattended Windows 8 or Windows 2012 setup system profile, you must use a computer running the web interface extension, where you have installed Windows ADK for Windows 8 in English under Windows Vista, Windows 2008, Windows 7, Windows 8 and Windows 2012. You cannot run this operation on a Windows 2000 or Linux operating system. The web interface extension must be started with local administrator privileges.
- If you want to update a cloning system profile, the disk content of the target you will use for the update will be deleted. Make sure you use a bare-metal target or a target with no valuable content on its disks.

1. Go to **Server** > **OS deployment** > **System profiles**. Double-click on a profile to view the details.
2. Click **Update** to open the update wizard.
3. Optionally, select **Update similar profiles** to update additional system profiles at the same time. Only system profiles compatible with the current one are available for selection. Unattended setup system profiles and WIM cloning system profiles cannot be updated together with cloning system profiles.
4. Follow the wizard instructions. Depending on the type of system profile, the wizard analyses the state of the target to ensure that all prerequisites are met. If all prerequisites are met, a new system profile is created, the old system profile taken as basis is kept. The name of the new profile is the name of the basis system profile with `(updated)` appended to it.

Browsing partition files

You can browse partition images stored on your server.

1. Go to **Server** > **OS deployment** > **System profiles**. Double-click on a profile to view the details.
2. In the **Original partition layout** section, click **Browse image of primary partition 1**.
3. You can expand or update the whole partition or a part of it.
   - To expand the whole or part of the partition:
     a. Right-click the folder you want and select **Expand on local disk**.
     b. Choose the computer where you want to expand and store the files contained in the selected partition.
     c. Specify the destination folder where to extract the partition files.
     d. Follow the instructions of the image wizard to expand the partition.

        Note: You must expand the partition to an empty directory. If you select a folder that is not empty the extraction fails.
   - To update the whole or part of the partition:
     a. Right-click the folder you want and select **Update from local disk**.

b. Specify the source folder of the OS deployment server where your updated data are located.

c. Specify the destination folder where to extract the partition files.

d. Follow the instructions of the image wizard to update the partition.

On the **Partition image explorer** page, you can create a new directory by selecting **Add new directory** in the contextual menu. You can also modify or add files by selecting **Upload file** in the contextual menu.

Note: File upload is limited to 16 MB.

Changing the partition layout

You can update partition layout to resize partitions, assign mount points, or change the file system.

If you change the partition layout in system profiles, you might render the profile unusable. It is recommended not to change the partition layout in system profiles, unless you know that the changes you want to make have no side effect.

In any case, do not transform a primary partition into a logical partition.

Note: Changing the partition layout from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose either one or the other entry point, and then perform all your changes from that entry point.

By editing the partition layout, you can:

- Add or delete partitions.

  Note: Adding or deleting partitions can lead to OS configuration problems. Use this feature carefully. To provide a clear description to your profile, use the **Comment** field.
- Resize a partition by dragging sliders, or by assigning it an absolute or relative size.
- Change the file system of a partition.
- Assign a mount point to the partition.

Note: When in the disk partitioning, both fixed sizes and percentage sizes are included, the exact reproduction is not guaranteed.

1. Click **Edit partition layout** on either the **Profile details** page or the **OS configuration details** page, **Disks** tab.
2.
   - To add a partition:

     a. Click **Modify partition layout**.

     b. Click into an existing partition.

     c. Click **Add a partition** in the contextual menu.

     d. Indicate the partition properties, including a mount point and click **OK**.

     Linux In a Linux profile, assign a mount point for the new partition. To be valid, this mount point must reference an existing directory in the main image. Starting from Fix Pack 3, the Linux profiles with the root partition as LVM are supported. In this case, you must ensure that the

HTTP mode is selected in the deployment scheme when deploying the profile. With the root partition as LVM, you cannot perform the deployment using the media.

- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

3.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

        Windows In a Windows profile, the operating system deployed using a system profile must be installed on the C: drive. Other drive letters are not allowed for the bootable partition.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

4.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

Modified partitions are aligned on megabytes rather than on cylinders. The following warning message might appear in the logs and can be safely discarded: `Warning: partition x does not end at a cylinder boundary`. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

If you want to use the same system profile with two different partition schemes, you can also duplicate a system profile by right-clicking the profile name and selecting **Duplicate profile**. The copy shares the same image files, but can have a different partition layout.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.
3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

   Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/cxdx`, where *x* is the disk number. Edit your device mappings accordingly.
4. Click **OK**.
5. Repeat step 2 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies unique.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Editing OS configuration parameters in the web interface for Windows operating systems

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.

4. Click **OK**.

Editing a Windows parameter file

You can modify OS configuration parameters by editing a file. This option allows you to modify parameters that are not displayed in the web interface. However, you must be experienced to use this option advantageously, because Tivoli Provisioning Manager for Images does not provide any syntax checking of the file. Information about the file format and syntax can be found in the documentation of the operating system itself.

1. To edit the file click **Edit custom 'unattend.xml'** on Windows Vista, Windows 2008, Windows 7 and click **Edit custom 'sysprep.inf'** on Windows XP, Windows 2003.
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). Unless otherwise specified, parameters specified in the default file override the content of the custom file.

Vista 2008 Windows 7 Content of this custom file overrides the default one created for the following parameters, which are written as tags:

- `NetworkLocation`
- `ProtectYourPC`
- `PersistAllDeviceInstalls`
- `UILanguage`
- `SystemLocale`
- `UserLocale`
- `InputLocale`

Content of the custom file is integrated within the default file for the following parameters: Value 1 is used in the default file and must not be used for these tags and attributes combinations in the custom file.

- `RunSynchronousCommand`, which takes a daughter tag `<order>`*`value`*`</order>`
- `LocalAccount`
- `Interface`
- `PathAndCredentials`, with the attribute `keyValue="`*`value`*`"`
- `DomainName`, with the attribute `keyValue="`*`value`*`"`
- `IPAddress`, with the attribute `keyValue="`*`value`*`"`

Only one tag `<component>` with attribute `processorArchitecture` is allowed. For all other tags, the values of the default file created override what the user has written in the custom file.

Troubleshooting:

If the OS configurations in the deployed operating system are not what you expected, you must examine carefully the parameter files. They are the result of the merge between the custom file and the default file created. See the log file `Windows/Panther/unattendGC/setupact.log` for problems in the file merge.

Note: Ensure you specify the full paths for the commands you use in the `unattend.xml` file.

Vista 2008 Windows 7 To troubleshoot OS configuration parameters after a successful deployment, view the two files `Windows/panther/setup.xml` and `Windows/panther/unattend.xml` which are the result of the merge between the default and custom parameter files. To troubleshoot OS configuration parameters after a failed deployment, you must look for the following files in the partition containing the operating system:

- `user_unattend.xml`, which is the file you edited
- `setup.xml`, which results from the merge
- `unattend.xml`, which results from the merge as well

XP 2003 To troubleshoot OS configuration parameters after a failed deployment, you must look for `$WIN_NT$.~BT\winnt.sif` in the partition containing the operating system. This file contains the information merged from the custom and the default files.

Binding drivers to a Windows system profile

When a system profile does not contain the drivers needed for deployment, you must bind these drivers to the system profile to be able to deploy it and obtain a working operating system.

If you encounter problem with the built-in drivers contained in your system profile, if some drivers are not bound, or if some drivers are missing, you should bind other drivers to your system profile.

You can only bind drivers to your system profile that are software modules in your OS deployment server. You must therefore create driver software modules from the drivers that you want to bind to your system profile.

Note: There are two methods to bind driver software modules to a system profile:

- the *standard binding rule method* where you can indicate profiles to bind to a software module.
- the *driver specific binding rule method* where you bind drivers per system profile and target model/device pair.

You can switch from one method to the other. In the *driver specific binding rule method*, driver bindings from the *standard binding rule method* are ignored, and vice-versa.

The method described here is the *driver specific binding rule method*.

From version 7.1.1.3 of the product onwards, it is recommended to use the *driver specific binding rule method*, which is the method by default on all new Windows system profiles.

The product helps you select appropriate drivers for particular target models. It helps you to predict potential problems and to solve them. It does not guaranty that a specific system profile, with bound drivers, works with a given target.

The information used by the OS deployment server to predict the compatibility of a driver with a target model is taken from the content provided by the vendor in its driver. The OS deployment server cannot verify the accuracy of this information.

1. Check the compatibility of your system profile.
   a. Go to **Server** > **OS deployment** > **System profiles**.
   b. To view the details of the system profile, you have two options.
      - Double-click on it.
      - Select a system profile, and then select **View profile** in the contextual menu.

   c. *(Optional)* In the section **Driver handling**, click **Switch to driver specific bindings mode**. You only need to perform this step if you are in the regular software binding rule mode.
   d. A check is performed while the page is loading. This may take a few minutes. By default, checks are performed on all available drivers.

      If drivers are missing, or are not bound, or if several drivers are bound for the same device, the following information is provided:

      ❌ Indicates a missing critical driver, or a critical driver of the wrong architecture.

      ✖ Indicates that a missing non-critical driver, or a non-critical driver of the wrong architecture.

      ⚠ Indicates that a required driver is present on the OS deployment server, but that it is not bound.

      ℹ Indicates that there are several drivers bound for the same device, or that there is a binding with a driver that is not known as compatible.

      You can expand the item to get more information.
      - For drivers missing on the OS deployment server, you can discover where to look for it, including, if available, a download link and the exact directory within the downloaded archive where the driver can be found.
      - When drivers are present on the OS deployment server, you can discover which driver to bind, in order of preference. If multiple drivers are known to possibly work for a device, the best choice is listed first. The choice is explained in the advice text, which first recommends the use of *device-specific drivers*, that is, drivers that have been specifically designed for the given hardware device. Then *compatible device drivers*, that match the device family, are recommended, even if they are not an exact rebranded variant (for example, as second choice, an Adaptec driver of the same family as an IBM ServerRaid adapter, if it is based on the same chipset). Finally, as third choice, *generic drivers*, for example, Microsoft generic AHCI driver for any AHCI controller, are recommended.

      If no error is found, you do not need to modify the bindings.
2. Modify the driver bindings of the system profile. There are two ways to perform this.
   - Use a wizard.
     a. Click **Fix Drivers**.
     b. Follow the instructions of the wizard. After having selected a target model, you have to select one of these options:

        Automatically fix issues which can be fixed for this model.
        Fixes all issues which can be automatically fixed. Such issues

include a missing binding to an existing driver, or multiple bindings for a device, for example.

Manually fix issues for this model.
Presents you with each issue in turn. Ways to solve the issue, when available, are proposed.

Automatically bind drivers for this model.
Erases every existing binding. New bindings are then automatically added.

Copy driver bindings for this model from a similar profile.
Copies all the bindings from a selected source system profile to the current one.

Reset all drivers bindings for this model.
Erases all the driver bindings, and does not create any new binding.

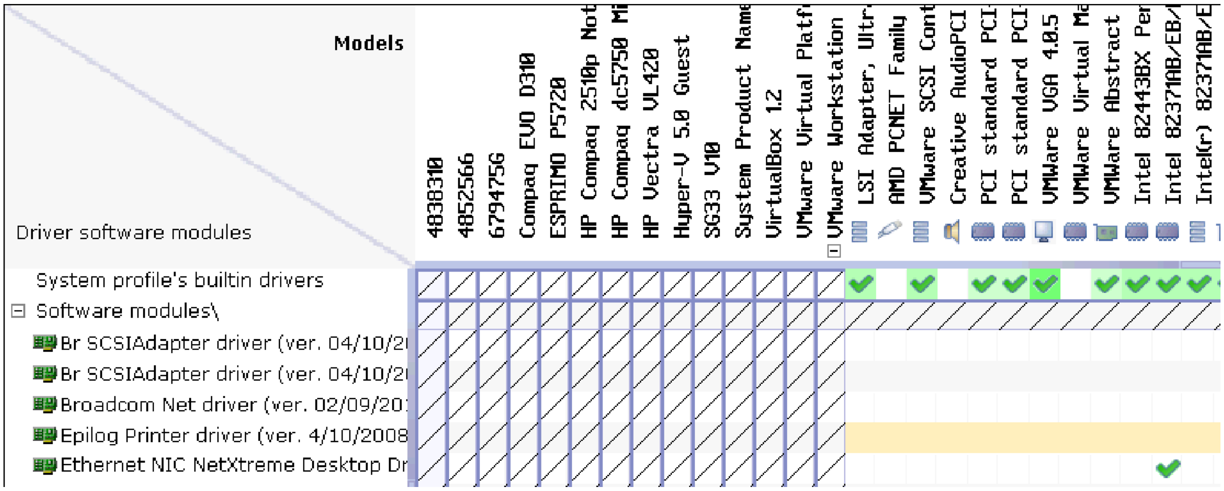
- Edit the bindings manually.
  a. Click **Edit profile's driver bindings** on the **Profile details** page.

     A grid is loaded.

     Columns represent target models known to the OS deployment server. They can be expanded to view their devices, provided an inventory has been performed.

     The first line represents the system profile. Other lines represent software module folders in the OS deployment server. They can be expanded to view individual drivers. If a driver can be used only for 32-bit or 64-bit machines, a superscript *x86* or *x86-64* mark is written next to the driver name. If you do not find the drivers that you need in the list provided, you should first create software modules for your drivers.

  b. Optional. To obtain a summary of the errors and warnings, click the link above the grid. This helps you locate the problematic areas in the driver grid.
  c. Expand the columns of problematic target models to view individual devices.
  d. Expand software module folders containing drivers to view the individual drivers.



Models

Driver software modules

4838310
4852566
6794756
Compaq EVO D310
ESPRIMO P5720
HP Compaq 2510p Not
HP Compaq dc5750 Mi
HP Vectra VL420
Hyper-V 5.0 Guest
SG33 V10
System Product Name
VirtualBox 1.2
VMware Virtual Platf
⊟ VMware Workstation
LSI Adapter, Ultr
AMD PCNET Family
VMware SCSI Cont
Creative AudioPCI
PCI standard PCI
PCI standard PCI
VMware VGA 4.0.5
VMware Virtual Ma
VMware Abstract
Intel 82443BX Per
Intel 82371AB/EB/
Intel(r) 82371AB/E

System profile's builtin drivers
⊟ Software modules\
Br SCSIAdapter driver (ver. 04/10/2
Br SCSIAdapter driver (ver. 04/10/2
Broadcom Net driver (ver. 02/09/20
Epilog Printer driver (ver. 4/10/2008
Ethernet NIC NetXtreme Desktop Dr

Figure 3. Driver binding grid

A cell with a green background indicates that driver information corresponds to the device. The quality of the drivers that can be selected is illustrated by the intensity of the green background: ■ ■ ■ the best drivers are in intense green, the family drivers are in standard green, and the generic drivers are in pale green.

A cell with an orange background indicates either that the driver is not a PCI driver, or that there is no compatibility information available for the driver.

A cell with a green check mark ✓ indicates that the driver is bound to the system profile for use with the specific target model and device.

e. Click on a green background cell to add or remove bindings.

   It is not possible to bind or unbind drivers from the system profile itself, because they are built-in drivers.

   You should have one, and only one, check mark per column, indicating that you have one and only one driver for each device.

f. When you are done modifying the bindings, click **Save**.

g. To return to the **Profile details** page, click **Back**.

   Potential problems with the image are recomputed, allowing you to check if your modifications have solved the detected problems.

When you have solved all the driver binding issues, you can deploy targets with your system profile.

Restoring a system profile manually

If you want to check that your cloning system profile contains all necessary information, you can restore it manually either from the web interface, or from the client computer.

Note:

1. You can only restore Windows cloning system profiles.
2. System profile restoration works only on targets of the same model as the one on which the profile was created. Restoring a profile on another model of targets might result in unexpected behaviors.
3. When you restore a system profile manually, the image is restored as-is, without any automatic parametrization. Thus, restoration cannot be unattended as some parameter values are required and must be entered manually.
4. Creating a cloned profile and restoring it manually is not meant as a backup procedure and it should not be used in that way.
5. A cloned profile can be restored on one target only at a time. Restoration cannot be performed on several targets together.

To perform the manual restoration:

1. Select the wanted target in the **Target Monitor**.
2. Select **Deploy now** in the contextual menu.
3. Follow the wizard instructions.

Restoring a profile from the web interface

1. Go to the **Target Monitor** page.
2. Select a single target
3. In the contextual menu, select **Additional features**

4. Select **Restore a profile**
5. Click **Next** and follow the instructions of the wizard.

Restoring a profile from the target

1. Click the icon to restore an image.
2. Click the **Restore a system profile** icon.

   Depending on the types of images on your OS deployment server, you can also get icons for

   - Restoring a software snapshot
   - Restoring a virtual floppy-disk
3. Select a system profile or a software snapshot from the list provided and click **Next**.
4. Optionally select options and click **Next** to restore the profile.
   - Windows If the system profile is a Windows image, it can include the Sysprep mini-setup wizard that is typically used to perform some post-configuration on the image.

     2000 2003 XP You can disable this mini-setup wizard for Windows 2000/2003/XP if you want to start the operating system and do some modifications before reinstalling Sysprep manually. In this case, a warning message appears, telling you that some minimal post-configuration are applied anyway, to avoid the risks of potential conflicts. This option is not available on Windows 2008/Vista.
   - Some computers can have been delivered with protected partitions for emergency restore backups. At this stage, the option is given to restore protected partitions or not.
   - Additionally, if a CMOS image was included in the system profile at the time it was created, you can decide whether you want to restore it. Remember that restoring a CMOS image on a target different (or with a different BIOS version) than the original can severely damage the target.

Generic System Profiles (GSP) for Windows operating systems

This section explains how to handle Generic System Profiles (GSP) for Windows operating systems.

Once created the GSP, if the binding flag is set to true, you can see the list of GSP that are available for deployment when you boot on the target.

Generating Generic System Profile

This topic describes how to use the rbagent command line to handle Generic System Profile.

To generate the Generic System Profile (GSP), in addition to the WIM file it is necessary to have a configuration file called <wim_name>.settings. The .wim/.settings pair must be copied under the product server files \files\global\[GSP-folder] where GSP-folder must be manually created and named GSP.

If the file <wim_name>.settings is in UTF8 encoding, in order to be recognized by the product, it must contain the first three bytes **`ef bb bf`**, also known as UTF8 BOM (byte order mark).

To generate, update or delete the Generic System Profile on the product server, use the following rbagent command line: **rbagent.exe –s <OSD_server_ip_address>:<OSD_server_passwd> rad-mkgenericsysprof**
where:
–s <OSD_server_ip_address>:<OSD_server_passwd>
provides information on the OSD server to be contacted, if not already present in the rbagent.conf.
The algorithm of the command line works as follows:

1. Scan the folder `\files\global\[GSP-folder]`, for `.wim/.settings` pairs. Only files having identical prefix names are taken into account.
2. Verify that all required parameters are available, if not generate an error and skip.
3. If a valid `wim/settings` pair is detected, create a new System profile/configuration in database and generate the complete rad-xxxx configuration file required by the product.
4. Optionally, if the section [Bindings] is defined and Rule= "true", a grouping Rule is created in the product database.
5. If a GSP profile is defined but there is no corresponding wim/settings pair, then `profile/configuration/bindings` are removed from the server database and configuration files (no deletion of the WIM or settings in GSP folder).
6. Optionally, if DriverMode= "auto", the Automatic Driver Binding functionality is enabled for the profile.

If a parameter in GSP settings is modified, the command line detects it and updates the `db/configuration/bindings` on the product server.

Configuration file

This section describes the configuration file to be used when you generate the Generic System Profile.

The configuration file must have the extension .settings and is divided into three main sections: [GSP],[Configuration] and [Bindings]. Parameters in every section may be mandatory (required during syspep phase on deployment) or optional.

Structure

`[GSP]` This section is mandatory and contains the following parameter:

- `Name`

`[Configuration]` This section is mandatory and contains the following parameters:

- `Ostype`
- `FullName`
- `OrgName`
- `AdminPasswd`
- `AdministratorName`
- `TimeZone`
- `Locale`
- `OSversion`
- `OSarchitecture`
- `Productkey`
- `DriverMode`

- RequestPasswd
- updatemenu

[Bindings] This section is optional and contains the following parameter:

- Rule

Parameters

The syntax of the parameters included in the configuration file is as follows:

- Name = [System profile name] - mandatory.
- Ostype = [Windows [XP|Vista|2008|2008R2|7 GSP] - mandatory.

  It is important to add the GSP suffix at the end of the OS type to identify a Generic System Profile.
- FullName = [Windows User Full name] - mandatory.
- OrgName = [Windows Organization name] - mandatory.
- AdminPasswd = [Administrator Password] - optional.
- AdministratorName = [Administrator Name] - optional.
- TimeZone = [Timezone ID code hex] - mandatory.

  The time zone is the standard numerical code that you find for example at http://msdn.microsoft.com/en-us/library/ms912391(v=winembedded.11).aspx. It must always be on three digits (adding zeros in front if necessary). For example, if you take the time zone code for Hawai, the code must be 002 and for W. Europe Standard Time the code must be 110.
- Locale = [Locale ID code hex] - mandatory.

  The locale ID must be the standard numerical HEX code that you find for example at http://msdn.microsoft.com/en-us/goglobal/bb964664. The code must always be on 4 digits (adding zeros in front if necessary). For example, if you take the locale for French-France, the code is 040c.
- OSversion = [OS name, Service Pack number, Build number] - mandatory.

  This parameter is split in three values separated by commas. The information related to the three values can be found in OS -> System information.
- OSArchitecture = [x86, x86-64...] - mandatory.

  This parameter is used to distinguish between 32 and 64 bit. Used values are x86 for 32 bit and x86-64 for 64 bit.
- Productkey = [Windows product key] - mandatory.
- RequestPasswd = [empty or plain text or CryptStr format] - optional.

  This parameter is used to protect the Generic System Profile (GSP) with a password. The CryptStr format is the result of the **rbagent rad-hidepassword** command without the **md5** option. If the string is left empty, no password is requested. If a password is specified, the deployment is performed only when you provide the right password. If the password entered is incorrect, no error message is displayed (only in the console).
- Rule = [true/false/empty] - optional.

  This parameter can support three values:

  Empty or not exists: nothing to do.

  true: create a grouping rule associated to the configuration.

  false: deletes all the grouping rules associated to the configuration.
- DriverMode = [none/auto] - optional.

This parameter is used to enable or disable the Automatic Driver Binding functionality. The default value is auto.

- `updatemenu` = [true/false] - optional.

  This parameter is used to flag all the binding menus on the target as "Out-of-sync" at every image change on the OSD server. The default value is true.

Example

Here is a typical example of GSP configuration file:

```
[GSP]
Name="GSP WIM Win7x64: Windows 7 Pro (x64)"

[Configuration]
ostype="Windows 7 GSP"
FullName="roro"
OrgName="IBM"
AdminPasswd="Password"
AdministratorName=""
TimeZone="110"
Locale="100c"
OSVersion="Windows 7 Professional, , 6.1.7600"
OSArchitecture="x86-64"
Productkey="FJ82H-XT6CR-J8D7P-XQJJ2-GPDD4"
RequestPasswd="password"
updatemenu="true"
[Bindings]
Rule="true"
DriverMode="auto"
```

The administrator password can be hidden using the web interface extension command **rad-hidepassword** without option md5. Open a command prompt in the same directory that contains the **rbagent.exe** executable file. On Windows operating systems, the path is generally `C:\Program Files\Common Files\IBM Tivoli\rbagent.exe`.

Run the **rbagent** command to encrypt your chosen password: **rbagent.exe –s <OSD_server_ip_address>:<OSD_server_passwd> rad-hidepassword <Password to encrypt>**

Your new encrypted password is generated and can be found in the Result string. For example, an encrypted password can look like the following one:

AA42154C94B0FD323231579210233951

Command line output

This section describes the output of the rbagent command line.

Every time a GSP command line is launched, it generates an output file called gsp.log in the GSP folder and provides information on whether the command generated a new Profile/Configuration, an update, a deletion or just an error.

For every wim/settings pair (named within curly brackets) the type of action is labeled and described:

<INSERT>: A new Generic System Profile and configuration is created on the product server.

<*UPDATE*>: A new Generic System Profile is updated (at least one parameter of the .settings file has changed).

<*REMOVE*>: The wim/settings pair no longer exists, the orphan GSP is removed together with the related configuration and binding.

<*NO ACTION*>: The wim/settings pair is detected but there are no differences with the existing GSP.

<*ERROR*>: The wim/settings pair is detected but a problem occurs in the .settings file structure (for example, a mandatory parameter is missing).

Here is a typical example of a gsp.log:

```
*** Generic System Profile --> rad-mkgenericsysprof command output ***

[2011/10/31 17:04:11] {Win7x64} <INSERT>: new profile in db and create
config/bindings

[2011/10/31 17:04:17] {Win2003x86} <UPDATE>: update gsp profile in db and/or
associated configuration

[2011/10/31 17:04:22] {WinVistax86} <ERROR>: parameter ostype is missing or
blank in the Configuration section

[2011/10/31 17:04:24] {WinXPx64} <INSERt>: new profile in db and create
config/bindings

[2011/10/31 17:04:26] {WinXPx86} <REMOVE>: remove profile in db and
associated config/bindings
```

Software modules for Windows operating systems

Software modules are images other than system profiles that can be created to address various needs.

Tivoli Provisioning Manager for Images is based on imaging technology. As administrator, you create images of components that you want to see on every target, and the automated deployment merges and restores these images on each target, automatically, when needed.

Tivoli Provisioning Manager for Images can handle most scenarios for software deployment and post-installation configuration.

Types of software modules

There are many types of software modules. Depending on the type of package and installation files, the wizard guides you through the different steps to achieve your software module with minimal effort. The types of software package supported by the wizard are listed in this section.

- Vista 2008 Windows 7 **Language pack**
- Vista 2008 Windows 7 **HotFix (MSU)**
- **A Windows application installation, using Microsoft Installer (MSI)**
- **A Windows driver to include in a deployment**
- XP 2003 **A Windows HAL to include in a clone deployment**
- **A custom action on the targets**. This includes OS configuration changes such as registry patches, commands to be run, and copying sets of files on the target.

WinPE and its uses

WinPE is widely used in all the tasks pertaining to the deployment of Windows operating systems. The product uses two different kinds of WinPE 3.x and WinPE 4.x, depending on the tasks at hand.

Types of WinPE

Windows Preinstallation Environment (WinPE) is a group of files which can be loaded as a ramdisk and which allows you to perform operations on a target.

WinPE 3.x and WinPE 4.x deployment engine

This WinPE is a prerequisite to create Windows system profiles and to deploy them.

To create a WinPE deployment engine, you need a computer running a Windows operating system, with Windows AIK for Windows 7 in English and Windows ADK for Windows 8 in English installed and running the web interface extension.

WinPE deployment engine creation always creates a 32-bit and a 64-bit deployment engines. In BIOS mode, the 64-bit WinPE 3.x deployment engine is used only to deploy unattended setup of Windows 2008 64-bit GA operating system. In UEFi mode, the 64-bit WinPE 3.1 deployment engine is used for all the tasks requiring a WinPE deployment engine.

WinPE hardware environment

This type of WinPE is used for hardware configurations.

To create a WinPE 3.x hardware environment, you need to start the vendor commands on a computer running a Windows operating system, with Windows AIK for Windows 7 in English installed, and the web interface extension running. You need to start the vendor commands before you start the web interface extension.

It is possible to create WinPE 1 and WinPE 2 hardware environments.

In BIOS mode, you must use the 32-bit WinPE 3.x, while in UEFI mode you must use the 64-bit WinPE 3.1.

Note: WinPE 4.x cannot be used to create a WinPE hardware environment,

WinPE2 ramdisk

WinPE2 ramdisks are obsolete from version 7.1.1.3 of the product onwards. You may keep those that were created with an earlier version of the product, or safely delete them. You cannot create new ones.

WAIK

Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

Note: Windows Automated Installation Kit for Windows Vista and Windows Server 2008 is not supported anymore. Use Windows Automated Installation Kit (AIK) for Windows 7 in English only.

You must restart your computer after having installed Windows AIK.

WADK

Windows Assessment and Deployment Kit (ADK) for Windows 8 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

Use it to provision Windows 8 and Windows 2012 operating systems. You must restart your computer after having installed Windows ADK.

Good practice

If you deploy Windows operating systems, you need to create 32-bit and 64-bit WinPE deployment engines, and potentially WinPE hardware configurations. For each of these creations, you need a computer running a Windows operating system , with Windows Automated Installation Kit (AIK) for Windows 7 in English or Windows Assessment and Deployment Kit (ADK) for Windows 8 in English and the web interface extension installed. The same configuration is also needed to update Windows Vista/2008/7/2012 /8 system profiles.

Windows AIK for Windows 7 in English can be obtained free of charge from Microsoft, but it is rather heavy and cumbersome to install. Therefore, it is good practice to install Windows Automated Installation Kit (AIK) for Windows 7 in English, and Windows Assessment and Deployment Kit (ADK) for Windows 8 in English, and the web interface extension on a dedicated computer running a Windows operating system and to perform all operations requiring this configuration on this dedicated computer.

If your OS deployment server runs under a Windows operating system, consider making your OS deployment server the dedicated Windows computer.

Creating software modules

There are distinct types of software modules which vary according to the operating system being deployed. The software wizard guides you through the creation of software modules for each type.

Creating a Language Pack software module

Windows Language packs can be created only from a computer with a Windows operating system and running the web interface extension.

The directory containing language pack files must contain a file with a `.cab` extension.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select **Windows Vista/2008/7/2012** and click **Next**.
4. Select **Language pack** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Note: The default Software Application Order of language packs "When the OS is installed" must not be changed.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

- A description that identifies the software module in the software module tree.
- A comment with additional information about the software module.
- A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
- The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a HotFix software module

Windows HotFixes can be created only from a computer with a Windows operating system and running the web interface extension.

The directory containing the HotFix files must contain a file with a `.msu` extension.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select **Windows Vista/2008/7/2012** and click **Next**.
4. Select **HotFix (MSU)** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a Microsoft Software Installer (MSI) software module

MSI software modules can be created only

- locally with a provisioning server installed on a Windows 2000/2003/2008 operating system
- from a computer with aWindows 2000/2003/2008/XP/Vista/7 operating system and running the web interface extension.

The directory containing MSI files must contain a file with a `.msi` extension. If the MSI file is located on the provisioning server, you must have placed it in a subdirectory of the `import` directory.

Note: If the folder you are looking for is not on the local computer, the provisioning server, or on another computer running the web interface extension, you might still be able to access the wanted resource using the following procedure:

Windows **Windows**

1. Create a `.lnk.yourfilename` file (where `yourfilename` is the name of your choice) that contains the path to the wanted folder (for example, `\\fileserver\export\softs\`).
2. In the wizard, enter `.lnk.yourfilename` preceded by the appropriate path.

To create your software module

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select **Windows Vista/2008/7/2012** or **Windows 2000/2003/XP** and click **Next**.
4. Select **A Windows application installation, using Microsoft Installer (MSI)** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

     For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=xh`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

     ```
     install /sPB /rs /rps /l <=1h
     ```
   - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with

the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a driver software module for Windows operating system

A driver package is used to provide the appropriate driver files to Sysprep or Windows unattended installation to install devices that are not activated by Windows because the driver is not present in the system profile.

The directory containing the driver files must contain a file with a `.inf` extension.

Driver packages are best used with unattended setup profiles, because standard Windows installation files do not always contain the drivers for recent hardware, and the goal of unattended setup is to have the target fully installed at the end of the process. However, driver packages can also be used with cloning-mode system profiles, because Sysprep can use driver packages to install new devices. There is no need to run Sysprep in `PnP mode` to have new devices installed when Sysprep runs on the target.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select **Windows Vista/2008/7/2012** or **Windows 2000/2003/XP** and click **Next**.
4. Select **A Windows driver to include in a deployment** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   When you indicate the directory in which the driver files are located, if several sub-directories contain drivers, the wizard lists all these directories. You must then select one or several directories. Selecting multiple directories allows you to create several driver packages at the same time with common binding rules.

   In case of multiple driver package creation, you can enter a folder name in which you want to store the new software modules. If the folder does not exist, the wizard creates it.

   If only one driver package is being created, the wizard presents the characteristics of the driver. This panel is skipped in the wizard in multiple driver package creation, but you can view the information in the software module details after the package has been created.

   The wizard allows you to create binding rules based on the PCI hardware ID, the baseboard ID, the computer model name, operating system architecture and targeted operating systems. Depending on your selections, the wizard provides steps with easy-to-follow instructions to create the binding rules.

   PCI hardware ID

   - If you select **Use this driver for the exact same device only**, the PCI vendor ID, device ID, and sub-device ID must match.
   - If you select **Use this driver for similar devices**, only the PCI vendor ID and the device ID must match.

   Baseboard ID

   You can either type in a substring of the baseboard name or select baseboard names extracted from the targets known to the OS deployment server.

Computer model name
You can either type in a substring of the computer model name or select model names extracted from the targets known to the OS deployment server.

OS architecture
Select **32 bit**, **64 bit**, or **Both** if you know the architecture the driver has been designed for. Select **Auto** to use the information contained in the driver to define the binding rule.

OS targeted
Select for which family of Windows operating systems the driver has been written for.

The number of rules created vary depending on the selections you made, but very quickly reaches over one hundred if your rules are based on similar PCI device IDs.

Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard for single driver creation. For multiple driver creation, the parameters are not displayed in the wizard. They can be edited in the software details page of each driver package. These parameters include:

- A description that identifies the package in the software module tree.

  Note: The pre-filled description might not be informative enough for you to know when you can use your driver. It is recommended to update the description and to include information such as operating system and architecture. You might need to use abbreviations because the description is limited to 50 characters.
- A comment with additional information about the software module.
- The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the package at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
- A file name to store your image on the OS deployment server. Packages typically have a `.pkg` extension.
- The path to where the installation files are restored on the target. This path must start with `\drivers`, because Windows unattended installation and Sysprep look in `C:\drivers` when installing new devices.

Importing drivers from the IBM Web site:

You can maintain your system device drivers and firmware at the most current levels avoiding unnecessary outages by using a new agent command based on the IBM UpdateXpress System Pack Installer tool.

This command creates a batch file that launches the IBM UpdateXpress System Pack Installer tool. You can use the batch file to detect current device driver and firmware levels, remotely retrieve the device driver and firmware updates from the IBM Web site, automatically package the drivers needed, and bind them to specific hardware models.

1. Run the following command to create the batch file:

   ```
   rad-mkuxspbatch uxsp-path dest-path (BOM | model=<type1>[,<type2>,...]
   OS=<OS1>[,<OS2>...,])
   ```

where:

uxsp-path
Specify the full path to the UpdateXpress setup utility.

dest-path
Specify the main path where all the updates are stored together with the UpdateXpress batch file.

BOM Specify the OS deployment server database BOM table to be scanned for detecting the updates needed.

model=
Specify the models used for manual updates by giving a list of model types (such as `model=4190,7971`).

OS= Specify the operating systems used for manual updates.

OSx Specify the operating system types: `windows`, `rhel3`, `rhel4`, `rhel5`, `sles9`, `sles10`, `all`.

A batch file `updatexpress.bat` is created in the directory `dest-path`. The generated batch file contains:

- Commands to acquire system packs and available updates (in particular drivers) for every model and operating system combination.
- Commands to extract every software module.

If you specified the BOM option, the agent command automatically scans the OS deployment server database for existing models and operating systems. You can also manually specify the models and operating systems to preload software modules of systems that currently do not exist within Tivoli Provisioning Manager for Images. You can run this batch file to import the drivers needed by using the IBM UpdateXpress System Pack Installer tool located in the directory `uxsp-path`.

2. You can then run this batch file: `dest-path\updatexpress.bat`, where `updatexpress.bat` is the name of the batch file containing the sequence of commands. It acquires and extracts the drivers needed by using the IBM UpdateXpress System Pack Installer tool located in the `uxsp-path` directory.
3. Use the Tivoli Provisioning Manager for Images web interface to create software modules of the drivers acquired and extracted in the previous step:
   a. Go to **Server** > **OS deployment** > **Software modules**.
   b. Click **New software** to run the software wizard.
   c. Select the relevant operating system and click **Next**.
   d. Select **A Windows driver to include in a deployment** and click **Next**.
   e. Select the computer and the main folder in which the driver files have been extracted (such as`..\IBM_Machine_type(7971)\OS_type(windows )`). The wizard lists all the drivers contained in this folder and its sub-folders.
   f. Select the drivers you need to package according to your hardware inventory and operating system.
   g. Specify the folder name where to store all the driver packages in a software tree structure.
   h. Select **Yes, create binding rules based on:** and then **PCI hardware ID** and **Target model name**. Click **Next**.
   i. Select **Use this driver for similar devices** and click **Next**.
   j. Specify the target machine model by selecting **the model name is one of the following** and then the model in the list.

k. For the chosen drivers select the appropriate operating system architecture (such as 32–bit) and the targeted operating system (such as `Windows Server 2003` or `Windows 2008`) and click **Next**. The driver packages are created with the specified binding rules and grouped in the folder you specified. You can also modify the binding rules by editing the software module or you can create an additional software module with other drivers and add it to the same main folder.

l. Click **Finish**.

Examples

Here is an example of generation of the `updatexpress.bat` file. It scans the BOM table to detect current Tivoli Provisioning Manager for Images device driver and firmware levels, remotely retrieves the device driver and firmware updates from the IBM Web site, and extract them into explicit model and operating system folders.

```
rad-mkuxspbatch d:\uxsp\uxspi300.exe d:\output BOM
```

Here is an example of generation of the `updatexpress.bat` file. It remotely retrieves the device drivers of model 4190 and 7971, for Windows, RedHat 3, and SLES 10 operating systems. It connects to the IBM Web site, to retrieve, package, and extract the drivers into explicit model and operating system folders.

```
rad-mkuxspbatch d:\uxsp\uxspi300.exe d:\output model=4190,7971
OS=windows,rhel3,sles10
```

Creating driver software modules for servers running Windows operating systems:

To deploy Windows operating systems on servers most efficiently, you need up-to-date drivers which are often not included with operating system installation files. These drivers can be obtained from the vendor of the server.

Before you can create your driver software modules, you must obtain the appropriate driver files.

IBM drivers
For IBM drivers, download the ServerGuide. To locate the ServerGuide, search for `ServerGuide download` in a search engine. Copy the `sguide` directory.

Note: ServerGuide is different from the ServerGuide Toolkit.

HP drivers
For HP drivers, download the SmartStart. To locate the SmartStart, search for `SmartStart download`.

Dell Drivers
To locate Dell drivers, search for `Dell drivers download`.

The following task assumes that the drivers have been copied into`Files/import` on your OS deployment server.

You might have to go through the driver software module creation process several times, to create different driver software module directories specific for operating systems and their architecture.

1. Go to **Server** > **OS deployment** > **Software modules**. Click **New Software**.
2. Select the relevant operating system and click **Next**.

3. Select **A Windows driver** and click **Next**.
4. Select **On the server itself (in the 'import' directory)** and click **Next**.
5. Select the relevant directory. For IBM drivers for a Windows 2003 operating system, this is `sguide/w2003drv/$oem$/$1/drv`.
6. Select all relevant drivers in the list provided and click **Next**. Sometimes, several versions of the same driver are available. In this case, follow these guidelines:
   - Select drivers without alternative, even if the name is misleading.
   - Select the appropriate Windows version when there are alternatives, for instance select *win2003* rather than *win2k* or *winnt* for a Windows 2003 driver.
   - Select *server* when the alternative is between *server* and *pro*.
   - Avoid selecting drivers with *powerpc* in their name.
   - Avoid selecting drivers containing hardware abstraction layer (HAL).
   - Avoid selecting drivers with another architecture.

   Note: It is better to have a few extra drivers included in the software module than to miss one.
7. Give a meaningful folder name to store your drivers, for instance `IBM ServerGuide 2003 32–bit`, and click **Next**.
8. Select **Yes, create binding rules based on:** and **PCI hardware ID**. Then click **Next**.
9. Select **Use this driver for the exact same device only** and click **Next**.
10. Select the appropriate architecture and the targeted operating system and click **Next**.
11. Click **Finish**.

Now, you can check that targets have the correct bindings.

Checking the drivers bound to a target:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Select the **Bindings** panel.
2. Check the OS configuration bound to the target. If it does not correspond to the operating system for which you just created drivers, you must switch to a more appropriate one.
   a. Select a OS configuration with the operating system you have just created drivers for.
   b. Go back to the Target Monitor.
   c. Double-click on the target.
   d. Select the **Bindings** panel again.
3. Make sure that there is only one disk driver. If there are several, you should delete outdated drivers from your software modules, modify the binding rules, or unbind the inappropriate driver from the target if it has been bound manually. To do so
   a. Click **Edit** in the **software bindings** header.
   b. Clear any unwanted driver and click **OK**.

Creating a WinPE driver software module for targets

When the WinPE deployment engine does not contain the drivers that you need for a specific target, you can inject these drivers into WinPE in a static way.

In the static driver injection process, you can only bind drivers, to your WinPE deployment engine, that are driver software modules in your OS deployment server. You must therefore create driver software modules from the drivers that you want to bind to your WinPE deployment engine.

You might have to go through the driver software module creation process several times, to create different driver software module directories specific for operating systems and their architecture.

The following task assumes that the drivers have been copied into`Files/import` on your OS deployment server.

1. Go to **Server** > **OS deployment** > **Software modules**. Click **New Software**.
2. Select the relevant operating system and click **Next**.
3. Select **A Windows PE driver** and click **Next**.
4. Specify the computer containing the drivers. You can select **On the server itself (in the 'import' directory)**, the local computer or another computer running the Web interface extension. Click **Next**.
5. Search the target vendor scripting toolkit for WinPE drivers. For example in the IBM ServerGuide Scripting Toolkit, look for a zip file with a name similar to `ibm_utl_tsep_2.00_winpe_i386.zip`. Extract the zip file, keeping the file structure. WinPE drivers are located under a path similar to `sgdeploy\SGTKWinPE\Drivers\WinPE_x86_2010-06-10\`
6. Select all relevant drivers in the list provided and click **Next**. Sometimes, several versions of the same driver are available. In this case, follow these guidelines:
   - Select drivers without alternative, even if the name is misleading.
   - Select the appropriate Windows version when there are alternatives, for instance select *win2003* rather than *win2k* or *winnt* for a Windows 2003 driver.
   - Select *server* when the alternative is between *server* and *pro*.
   - Avoid selecting drivers with *powerpc* in their name.
   - Avoid selecting drivers containing hardware abstraction layer (HAL).
   - Avoid selecting drivers with another architecture.

   Note: It is better to have a few extra drivers included in the software module than to miss one.
7. Give a meaningful folder name to store your drivers, for instance `IBM ServerGuide 2003 32–bit`, and click **Next**.
8. Select **Yes, create binding rules based on:** and **PCI hardware ID**. Then click **Next**.
9. Select **Use this driver for the exact same device only** and click **Next**.
10. Select the appropriate architecture on which your driver runs. In BIOS mode, 32-bit WinPE is used for all the tasks requiring a WinPE deployment engine. Basing on operating system to deploy for deployment tasks (WinPE4 for Windows 8/2012, WinPE3 for earlier operating systems) and matching model patterns. For non deployment tasks, for example, destroy hard disk content in kernel-free mode, or operating system capture, if both WinPE3 and WinPE4 model patterns match with the target model, WinPE4 is the preferred choice. 64-bit WinPE3 is used only to deploy Windows Vista 64-bit and Windows 2008 64-bit unattended setup system profiles. For these two operating systems, both architectures are required. In UEFI mode, 64-bit WinPE is used for all the tasks requiring a WinPE deployment engine.

11. Select the targeted **Windows PE3 / PE4** operating system and click **Next**
12. Enter the description of your software module and click **Next**.
13. When the software module is created, click **Finish**.

Heuristics to select drivers to work with a WinPE deployment engine:

Drivers compatible with a WinPE deployment engine are not necessarily the same as the drivers for an operating system. In any case, software modules must be created from the drivers before they can be used in the OS deployment server.

When you create driver software modules for use with a WinPE deployment engine, it is sometimes difficult to know which drivers work with WinPE. Here are a few heuristics to locate the appropriate drivers.

- Select drivers for the appropriate operating system architecture.
- Prefer monolithic drivers containing only a simple `.inf` file and `.sys` file (without a CoInstaller DLL). This is typically the case with drivers provided in a RIS package.
- Prefer drivers for Windows 7 and Windows Server 2008 R2 operating systems. If these drivers are not available, you can try drivers for other Windows operating systems.
- The driver must support the correct PCI device. For example, the PCI inventory for the target shows a network card with
  - VendorID: 1111
  - SubVendorID: 2222
  - DeviceID: 3333
  - SubDeviceID: 4444

  Then the drivers `.INF` files should include a line ending with `PCI\VEN_1111&DEV_3333` or a line ending with `PCI\VEN_1111&DEV_3333 &SUBSYS_44442222`.
- If you need Broadcom NetXtreme II drivers, you must get the drivers in the special RIS package.

If you group your WinPE drivers within the same software module folder, it is easier to locate them when you bind drivers to your WinPE deployment engine.

Creating a software module for HAL injection on a cloning system profile

2000 2003 XP

Hardware abstraction layer (HAL) can change from one computer to another depending on whether it has a single or multiple processors and on, whether it uses Advanced Programmable Interrupt Controller (APIC) and Advanced Configuration and Power Interface (ACPI). HAL also depends on the operating system. To create universal images, you might be required to have HAL versions on your system profile different from the original.

To create a HAL software module to be injected on a cloning system profile during deployment, you must:

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New Software**.
3. Select **Windows 2000 / 2003/ XP**.
4. Select **A Windows HAL to include in a clone deployment**.

5. Follow the wizard instructions. Different HALs are available on Windows installation CDs. The wizard offers you to create binding rules for this HAL and pre-fills some of the data to facilitate the rule creation process.

If you did not use the wizard to create binding rules, it is recommended that you bind your HAL package now to deploy it in appropriate contexts.

Creating a software module for HAL injection on an unattended setup system profile:
HAL injection on an unattended setup system profile is typically only necessary on some very specific server systems. The server vendor must then provide you with the appropriate HAL. IBM provides HALs for its servers on the ServerGuide CD

To create a HAL software module to be injected on an unattended setup system profile during deployment, you must create a HAL software module and bind it to the corresponding system profiles.

To do this:
1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New Software**.
3. Select **Windows 2000 / 2003/ XP**.
4. Select **A Windows driver to include in a deployment**.
5. Follow the wizard instructions. When asked for the driver file location, provide the path to the HAL.

If you did not use the wizard to create binding rules, it is recommended that you bind your HAL package now to deploy it in appropriate contexts.

Creating a custom action software module for Windows operating systems

Software modules can also contain custom actions to be performed on the target. They are divided into:

- Vista 2008 Windows 7 A WinPE 2.0 ramdisk image

  Note: To create a WinPE2 ramdisk, the web interface extension must be started with local administrator privileges.
- 2000 XP 2003 A WinPE 1.5 ramdisk image
- An OS configuration change to perform on the target
- A set of files to copy on the target

Configuration changes are further subdivided into:
- Copy and run a single file
- Apply a Windows registry change
- Apply a Windows `.ini` file change
- Copy a single text file
- Execute a single command file
- Boot a virtual floppy disk

  Note: Virtual floppy disk software modules can only be created from a Windows operating system running the web interface extension.

In the OS configuration change wizard screen, you can select **Activate keyword substitutions**. If you use this option, you can specify which keywords must be substituted in the software module details, as described in the "Keyword substitution" on page 73.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select the operating system and click **Next**.
4. Select **A custom action on the target** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The full path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

     For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=xh`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

     ```
     install /sPB /rs /rps /l <=1h
     ```

   - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

As examples are described the complete step-by-step process of creating a software module with the content of the second CD of a Windows 2003 R2 distribution (see "Creating a software module for unattended deployment of Windows 2003 R2 operating system" on page 70), and of creating a ramdisk from a bootable diskette (see "Creating a ramdisk software module from a bootable diskette" on page 70).

Repeating custom actions:
Some commands must be run every time the target boots during a deployment. This is typically the case if you want to repeatedly connect a network share. This

connection is destroyed when rebooting. You can therefore create a single software module with a `netuse` command to set the network share and set this software module to run once after each reboot, starting at a specific reboot.

This option is available for

- Windows registry changes.
- Copying and executing a single file.
- Executing a single command.

1. Create your software module.
2. Double-click on the software module name in the **Software components** page to obtain the **Software details** page
3. Click **Edit** in the title of the **Package information** section.
4. Select the installation stage at which the software module must be applied first.
5. Select **Run at each software pass until end of deployment** and click **OK**.

Creating a software module for unattended deployment of Windows 2003 R2 operating system:
To prepare an unattended deployment of Windows 2003 R2, you must include some of the content of the second CD of the distribution in a software module and bind this software module to the system profile created with the first CD.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **Windows 2000 / 2003 / XP**.
4. Select **A custom action on the target**.
5. Select **A set of files to copy on the target (with an optional command to execute).**
6. Indicate on which computer the files of the second CD are located.
7. Indicate the complete path to find the files in `/CMPNENTS/R2`, for example `D:/CMPNENTS/R2`.
8. Verify the proposed description and if necessary, modify it. Optionally, enter a comment.
9. Enter the necessary parameters for this specific software module:
   - Apply the software module **After one additional reboot**.
   - Enter a meaningful package file name, with a `.pkg` extension.
   - Use `\install\R2` as destination path
   - Do not forget the command-line to be run on the target

     ```
     cmd /c \install\R2\setup2.exe /q /a /p:xxxx-xxxx-xxxx-xxxx-xxxx /cs
     ```

     where `xxxx-xxxx-xxxx-xxxx-xxxx` is the product key.
10. Wait during the package generation process and click **Finish**.

Do not forget to bind your software module to your Windows 2003 R2 unattended setup system profile.

Creating a ramdisk software module from a bootable diskette:
Creating a ramdisk software module from a bootable diskette is considered by the software module wizard to be a **Configuration change**, which itself is included in the **Custom action**.

1. On the **software modules** page, click **New software**. This opens up the software wizard.

2. Select **Windows 2000 / 2003 / XP**.
3. Select **A custom action on the target**.
4. Select a **Configuration change**
5. Select **Boot a virtual floppy disk**.
6. Specify which computer the bootable diskette must be read from. This can be either on the local computer or on another computer running the web interface extension . The option **On the server itself** must not be used.

   Note: If the diskette drive is added after the web interface extension is started (on the local or remote computer depending on your choice), it can be necessary to stop and restart the web interface extension before it can detect the diskette drive. Moreover, the diskette must not be opened by another application (such as Windows Explorer) as this can cause interference.
7. Insert the bootable diskette that you want to image and run as a ramdisk in the disk drive and click **Next**.
8. Enter a software module description and click **Next**.
9. Specify parameters for the package creation and click **Next**. The software module is created.

Creating a software group

Simplify the management of your software modules by grouping them into containers called *software groups*.

A *software group* is a collection of software modules that behaves as a standard software module.

The advantage of software groups is to manipulate only one object instead of several software modules when they should all behave in the same way. For example, you can select a whole software group for deployment, create a binding rule for it, or change its software application order, instead of doing it for each software module individually.

The elements of a software group are individual software modules. You cannot nest software groups within software groups.

A software module can belong to several software groups simultaneously.

To create a software group:
1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **A software group** and click **Next**.
4. Select all the software modules that you want to include in your software group and click **Next**.
5. Follow the remaining instructions of the wizard to create your software group.

You can now create binding rules for your software group, modify its application order, export it to a RAD file, or use it in a deployment, as if it were a standard software module.

You can also edit the software group, for example to add or remove software modules.

Editing software modules

You can edit the basic parameters of a software module, upload new files into your software module, and update drivers.

1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
2. From **Software details** page, use the links and buttons.
   - To edit the base parameters of a software module, click **Edit** at the top of the **Software module information** section.
   - To update files or add new files into the software module, click **Edit software module files**, or a link with a similar name, and select **Upload file** from the contextual menu.

     Note: File upload is limited to 16 MB.
   - For software groups, to add or remove software modules:
     a. Click **Edit** at the top of the **Software group contents** section.
     b. Select the software modules that you want to add.
     c. Deselect the software modules that you want to remove.
     d. Click **OK**.

Keeping command lines confidential

When you use command lines in your software modules, their call and their output are stored in deployment logs. In some circumstances, for example when the command line includes a password or a product key, it might be necessary to keep the information contained in the command line confidential. Three levels of confidentiality are available.

No confidentiality

The command line is visible in the web interface and on the target during the installation, its call is logged, and its output is also logged.

The command line call is not logged

The command line is visible in the web interface, and its output is logged, but the command line call, containing the whole command line string with all parameters, is visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by one exclamation mark (`!`).

The command line call and output are not logged

The command line is visible in the web interface, but its call and output are visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by two exclamation marks (`!!`).

To keep command lines confidential:

- Enter the appropriate number of exclamation points in front of the command in the Software Wizard when first creating the software module.
- Edit the software module information
  1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
  2. Click **Edit** in the Software module information banner.

3. Update the command line with the appropriate number of exclamation points.
4. Click **OK**.

Keyword substitution

You can usefully use keyword which act as variables and are substituted with their values during deployments. Keywords can either refer database values or server specific values, given by the user.

Syntax

Variable substitution expressions follow the syntax given here. They start with the character { and end on the same line with }. Words between these two characters are interpreted by using one of the following schemes:

- *{$expr$}* the expression is replaced with the string resulting of the evaluation of `expr`.
- *{/expr/ab}* the expression is replaced with the string resulting of the evaluation of `expr`, but each occurrence of the character "`a`" is replaced by the character "`b`" (character-based substitution).
- *{=expr=test content=this is a test}* the text "`this is a test`" is included in the destination file only if the string resulting of the evaluation of `expr` is equal to the text "`test content`".
- *{!expr!test content!this is a test}* the text "`this is a test`" is included in the destination file only if the string resulting of the evaluation of `expr` is not equal to the text "`test content`".

Note: If a variable does not exist (for example, it contains a typing error or it is not described in `server.ini`) but it is used in a command, its value is supposed to be empty which can result in deployment errors.

Database keywords

Within an expression, database records can be referred to. Within a record, each field can be accessed using the standard C notation (`record.fieldname`). The exhaustive list of these fields can be obtained from the database records, with the following correspondences between variable and database record names:

Table 3. Records for `free-text conditions`

Variable record name	Database record name
Disk	DiskInventory
DMI	DMIInventory
Order	BOM
User	UserProfile
System	SystemProfile
PCI	PCIInventory

Below are a few examples of available fields:

- `Order.IP`: a string, the target IP address, such as `192.168.1.2`
- `Order.MAC`: a string, the target MAC address, such as `00:01:02:03:04:05`
- `Order.SN`: a string, the target Serial Number, such as `CH12345678`

- Order.Model: a string, the computer model name, such as e-Vectra
- User.UserCateg0: a string, without any restriction, such astechnicians
- DMI.Vendor: a string, the vendor name, such as Hewlett-Packard
- DMI.Product: a string, same as Order.Model
- DMI.ProcModel: a string, the processor model
- Disk[0].Type: a string, the disk 0 drive type, such as ATAPI
- Disk[0].Media: a string, the disk 0 media type, such as Disk or CD
- Disk[0].DiskSize: a number, the physical size of the disk (if detected)
- PCI[0].VendorID: a string, the hexadecimal vendor ID of the device
- PCI[0].DeviceID: a string, the hexadecimal device ID of the device

For disks and PCI devices, you can use the function sizeof (sizeof(Disk) and sizeof(PCI)) to discover the number of devices present. You can then use indexes to access these devices.

As an example for keyword substitution, if BomID has OrgName Rembo SaRL, RemboServer 192.168.168.16, and IP 192.168.168.32 for value 1, the following text

```
BomID:{$Order.BomID$}
OrgName:{$User.OrgName$}/{$StrToLower(User.OrgName)$}
RemboServer:{$Order.RemboServer$}
IP:{$Order.IP$}
```

gives the following results after keywords are substituted (note the use of a Rembo-C function within the expression to be substituted):

```
BomID:1
OrgName:Rembo SaRl/rembo sarl
RemboServer:192.168.168.16
IP:192.168.168.32
```

Server specific keywords

If you want to set up server specific keywords, which are defined exclusively by the user and per server, you must edit Files/global/rad/server.ini.

Start the file with [Custom] and add a line per keyword, in the format **keyword=value**, where keyword is a word of your choice and value the value you want to give it.

To use the keyword in a command, type Server.keyword and activate keyword substitution when creating the software module.

Note: server.ini is not replicated between servers. If you use multiple servers, you must edit server.ini on each server.

Customizing the software page

You can view the software modules in a tree viewer or in a list viewer. The list viewer allows you to customize the visible information.

You must have created at least one software module, otherwise there is nothing to view.

To customize the visible information

1. Go to **Server** > **OS deployment** > **Software modules**. Then click **List view**.

2. From the list view, you can
   - Drag the column separator in the column heading to resize the column.
   - Click on the triangular arrow to the left of the column name to sort the software modules by column criteria.
   - Click on the arrow on the right of the column name and select an option to filter the information. Filtering on several columns is cumulative.
3. For more options, right click anywhere to open the contextual menu and select **Arrange columns**.
   - Select the columns you want to see and clear the others.
   - Click on the minus or plus icons to decrease or increase the size of a column.
   - Select a column and use the up and down arrows to move the column relatively to the others.

   Click **OK** to save your changes. The updated version of the list view is visible in the **Software modules** page.

To return to the tree view, click **Tree view**. You can also access the details of the software modules by double-clicking on a software module name, from either view.

OS configuration and software bindings

OS configuration bindings determine which configurations are available to a target when booting the target on the network, while software bindings correspond to the list of software modules currently assigned to the target.

OS configuration and software bindings are created when:
- The Target Monitor has been used to manually modify OS configuration and software bindings for the target
- A deployment has been started with the Target Monitor. In this case, an OS configuration binding is added for the corresponding OS configuration.
- Automatic binding rules are configured in the **Details** page of OS configurations or software modules. Some of these rules have matching values for the specified criteria. These bindings cannot be modified, except by modifying the rules.

With the Target Monitor, you can browse, remove or add OS configuration and software bindings to any target present in the database. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Binding software modules and OS configurations to targets

Bindings link software modules and OS configurations to targets to enable automatic deployment. When binding to targets, you explicitly provide the list of software modules and OS configurations to bind to your target.

To explicitly bind a software module or a OS configurations to a target, there are two methods:
- From the **Target Monitor** page
- From the **Target details** page

If you want to bind software modules or OS configurations to a group of targets, you must do it through the Target Monitor.

From the Target Monitor:

1. Select a target or a group of targets

2. Select **Bind software** or **Bind OS configurations** from the contextual menu
3. Select the items to bind from the popup window
4. Click **OK**

From the Target details page:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.
2. Go to the **Bindings** panel.
3. Click **Edit** in the relevant section to add explicit bindings for OS configurations and software modules.
4. Select the items for which you want to add explicit bindings.
5. Click **OK**

You can also clear items to remove their explicit bindings. To remove a binding by rule, you must modify the rule.

Binding software modules to a deployment scheme

Software modules can be bound to deployment schemes.

Take a company with offices in three locations: New York, Quebec City, and Mexico City. In each of these locations, the company has people in human resources, sales, logistics, and product development. For the sake of simplicity, consider further that all the employees use either one of two types of computers: a desktop, or a notebook. All desktop computers are identical (with the same network card, system board, disks, and so on) and the same applies for all notebooks.

In this scenario, the company needs two profiles, one with the image for notebooks and one with the image for desktop computers. Three configurations per profile (six in total) are necessary to integrate the different parameters of the different locations, in particular language and time zone information. Finally, schemes are set according to the employees' department, with software modules specific to the different departments bound directly to the deployment schemes.

1. Go to **Server** > **OS deployment** > **Task templates** Select the **Deployment Schemes** folder. Double-click on a deployment scheme to view its details.
2. Click **Edit** on the **Software bindings** section of the page to open the dialog to bind software modules to schemes.
3. Select which software modules you want to bind to your deployment scheme, in addition to software modules that can have been bound to targets.
4. *(Optional)* If you want to use only the software checked in the window when deploying with this scheme, select the **Discard all other software binding rules** check box.

Automatic binding rules

Automatic binding rules are used to create bindings between OS configurations and targets, or software modules and targets, without having to specifically bind a OS configuration or a software module on each target.

Rules are created in OS configurations and software modules to determine which targets are automatically bound to the OS configuration or software module.

Rules are made of criteria and values. If a target has a matching value for all criteria in the rule, the OS configuration or software module will be bound to that target. The binding will be displayed with the mention **by rule** in the OS configuration panel of the target properties for targets that match the criteria. For example, if the criteria is the model name, and the value is `Optiplex`, targets with a model name starting with `Optiplex` will be bound to the object where the rule has been defined.

Automatic binding rules are defined in Tivoli Provisioning Manager for OS deployment at the bottom of the **OS configuration details** or **Software details** page.

To create a new binding rule, click **New rule** located at the bottom of the Web interface:

1. The dialog displayed to create a new binding rule is different depending on whether you are adding a rule to an OS configuration or to a software module. When adding a binding rule to a software module, you can set values for the following criteria:
   - A deployment scheme
   - A system profile
   - A current OS configuration
   - Administrative group
   - One of the system-definable and user-definable fields of the database (only used if you have customized the database)
   - An operating system type, such as Windows 2000
   - An operating system version, such as SP2
   - An operating system language
   - An operating system architecture, such as x86-32
   - A computer model name
   - A BIOS version
   - A PCI device
   - A base board
   - MultiChassi
   - HAL Type
   - A free-text condition in `Rembo-C;` syntax

   For example, to create a binding based on the operating system type between a software module and targets, you must create a new rule, click **OS type**, and select the operating system version that you want to limit this software module to.
2. When adding a binding rule to an OS configuration, you can set a condition on the deployment scheme, and on the computer model name. The next ten fields are only used if you have customized your database and want to match specific user categories.
3. Finally, you can enter a free-text condition following the `Rembo-C;` syntax. They must only be used by advanced users.

   The conditions determine the applicability of the rule and evaluate to `true` or `false`. A condition must be formed using the variables also used for keyword substitutions in software modules, combined with Java™-like logical operators, listed by order of priority in the table:

Table 4. Logical operators for free-text conditions

Operator	Meaning
`<`	smaller than
`<=`	smaller than or equal to
`=>`	greater than or equal to
`>`	greater than
`==`	equal to
`!=`	not equal to
`&&`	AND operator
`||`	OR operator

For example, a typical condition can be:

```
Disk[0].DiskSize > 10*1024*1024
```

Note: If a condition cannot be evaluated, it is considered to have the value `false`.

Scheduling the application of software modules for Windows operating systems

Tivoli Provisioning Manager for Images provides a wide flexibility in the specification of a deployment task. As several software modules can be deployed in conjunction with a system profile, you can schedule when they must be applied.

Windows Preinstallation Environment (WinPE) is a prerequisite to create Windows software modules and to deploy them.

Tivoli Provisioning Manager for Images provides a wide flexibility in the specification of a deployment task. As several software modules can be deployed in conjunction with a system profile, you can schedule when they must be applied.

Typical application locations for software modules include:

- Sysprep and unattended setup processes are automatically run during the OS installation phase, if required.
- For other software: when the OS is installed or after additional reboots depending on the software module needs.

Software modules are not ordered within an installation stage. If you want a software module to be installed before another between two specific reboots, create two distinct installation stages between the reboots. For example, if your first software module copies files on the target and the second one runs a command on these files, you must place the first software module in an installation stage which occurs before the one in which you run the command software module.

1. To schedule the application of software modules, go to **Server** > **OS deployment** > **Software modules**. This opens a dialog window that allows you to order the different software modules stored on your OS deployment server. The dialog shows the different steps of a deployment with disk partitioning (in green), OS installation (in purple) and reboots (in red). Software components can be installed in between all of these steps, where they are placed inside the expandable installation stages (in yellow).
2. You can add, move, and delete reboot sequences by using the buttons at the bottom of the dialog window. You can also rename software installation stages.

3. You can expand the software installation stages to view their content by clicking on the **+** icon. You can then move individual software modules from one stage to another by drag-and-drop. The destination stage does not need to be expanded.

   Note: Drag-and-drop is limited to the **Software Application Order** window. You cannot drag-and-drop an item from the Software Module page.

   Note: Vista 2008 Windows 7 If you have more than one HotFix (MSU) software module in stages occurring later than **When the OS is installed**, you must ensure that they each have a different destination path on the target.

Starting from 7.1.1 Fix Pack 6 the capability to run a software module has been added before **Disk partitioning** and before **OS installation**.

Before partitioning the disk you can:
- Set the working directory on other drive or network share.
- Get information about the previous installation.
- Update firmware.
- In general, perform any other action that you want to run under the WinPE environment before the hard disk is altered.

Note: The following limitations apply to Windows software modules run before partitioning the disk:
- Working drive is `X:`, software with relative paths are copied on the `X:` drive. It is not possible to copy them directly on `X:` because of space limitation and also because this is done in a different deployment phase than the one in which the command is running. Files on `X:` would be lost. It is, therefore, important to specify the absolute path for the command line to run, for example c:\install\*folder**command line*.
- No more than 32 MB can be written to the `X:` drive (WinPE limitation).

Before installing the operating system you can:
- Use it mainly for workaround and debugging purposes.

Note: The following limitations apply to Windows software modules run before installing the operating system:
- Working drive is always `X:`, software with relative paths are copied on the hard disk.

When creating a recovery CD or exporting a RAD file, the software application order is automatically included.

Working with hardware configurations

It is sometimes necessary to run configuration tasks on the targets before installing the operating system, for example to update the firmware or to configure RAID volumes.

To automate this kind of operation with the product, you must perform a *hardware configuration task*, which uses a *hardware configuration object* stored on the OS deployment server. To create a hardware configuration object, you must have already created a *hardware environment*. This hardware environment contains WinPE or DOS files, updated with drivers specific to given hardware models and vendor-specific tools to perform hardware configuration tasks.

The hardware configuration tasks that you can perform with the product are

- RAID configuration
- Firmware update (BIOS and UEFI)
- Firmware settings (BIOS and UEFI)
- Hardware custom configuration, that is, any kind of tool that you can load into the environment and run from a command line.

You can also perform an inventory of RAID or Fiber Channel hardware.

Hardware configuration tasks are available only for targets with an x86 or an x86-64 architecture.

Example

To configure hardware with the product, for example a BIOS update with WinPE2 on an IBM target, you need to follow a number of steps.

1. Create a hardware environment with drivers and tools:
   a. Download Windows Automated Installation Kit (WAIK) from Microsoft and install it to have the WinPE2 files available.
   b. Download the latest ServerGuide scripting toolkit from IBM and extract it, for example, in directory `C:\IBM-SGTSK-WinPE2.x`.
   c. Run the **`SGTKWinPE.cmd`** command to prepare the WinPE2 environment with the needed IBM drivers. It creates the `.\sgdeploy\WinPE_ScenariosOutput\Local\RAID_Config_Only\ISO` directory, which contains both the WinPE2 binaries and the vendor-specific tools.
   d. Create a hardware environment with the hardware environment wizard.
2. Create a hardware configuration object with the hardware configuration wizard:
   a. Select **BIOS update** as the type of hardware configuration to be performed.
   b. Associate the hardware environment of step 1 and your hardware model to the new hardware configuration object you are creating.
   c. Indicate the location of the BIOS update material, that is, a set of files containing in particular `wflash.exe`.
3. Perform the actual configuration task by deploying the hardware configuration object of step 2 on your target:
   a. Select a target (or several) in the Target Monitor.
   b. Select **Deploy now** in the contextual menu.
   c. Select **Perform hardware configuration tasks** and optionally other deployment tasks in the deployment wizard.
   d. Select the hardware configuration object that you want to apply and follow the remaining instructions of the wizard.

The hardware environment now runs as a ramdisk on the target, and, using vendor-specific tools, the BIOS is updated.

Setting up your environment

To perform hardware configuration tasks, you must set up a hardware-specific environment containing the vendor-specific scripting toolkit tools and the necessary drivers to run correctly (for example, network connectivity) on the target.

The hardware environment supported are those running scripts and tools in:

- WinPE 3.x
- WinPE 2.x
- WinPE 1.x
- DOS

Every environment is very specific to its vendor, and must be prepared with the suitable drivers and scripting toolkit tools.

WinPE3, WinPE2, WinPE1, and DOS cannot perform hardware configuration tasks (for example, RAID configuration or BIOS setting) by themselves. They must contain drivers to access the hardware and tools to perform the configurations. These drivers and tools are vendor-specific and vary for each type of target model. When you create an environment with the OS deployment server, you associate either WinPE3, WinPE2, WinPE1, or DOS, to vendor-specific drivers and tools. You can then associate the resulting environment to a specific set of target models and a type of hardware configuration tasks to create a hardware configuration object.

Because a hardware environment is run as a ramdisk, it does not leave any trace on the target after the hardware configuration task is performed.

64-bit WinPE 3.1 is required for performing hardware configurations tasks in UEFI mode.

Hardware configuration objects and tasks

A hardware configuration object is the association, on an OS deployment server, of a vendor-dependent environment, target models, a type of hardware configuration to be performed, and possibly some other commands. A hardware configuration task is performed at deployment time by loading and running the associated hardware configuration object containing a vendor-dependent environment on the target, before installing the operating system.

Hardware configurations tasks do not impact the following operating system deployment because Tivoli Provisioning Manager for Images configures the hardware through actions run in a ramdisk before the deployment of the operating system.

The execution flow is similar, regardless of the environment to run, or the type of hardware environment task:

1. The environment is loaded in memory, as a ramdisk
2. Any additional binary or configuration files are added to the ramdisk, based on the selection made in the web interface when creating the hardware configuration object
3. The computer boots the ramdisk
4. The hardware configuration task is run
5. The computer reboots

6. Tivoli Provisioning Manager for Images resumes the deployment sequence if any was selected, but a hardware configuration object can be run also as an independent task

The following types of hardware configuration objects are available:

RAID configuration
The hardware configuration wizard allows you to create a hardware configuration object to configure RAID adapters in a vendor-independent way. Tivoli Provisioning Manager for Images builds the vendor-specific configuration file.

BIOS update
The hardware configuration wizard allows you to create a hardware configuration object to update the BIOS firmware on the target.

BIOS settings
The hardware configuration wizard allows you to create a hardware configuration object to update the BIOS or BMC (baseboard management controller) settings through an initialization file.

Hardware custom configuration
The hardware configuration wizard allows you to create a hardware configuration object to perform any kind of hardware configuration. Any tool used for preparing the environment can be packaged in a custom hardware configuration object, injected into the ramdisk and run using command lines.

Capture hardware parameters
This option is available only if you do not already have a hardware capture configuration object.

The hardware configuration wizard allows you to create a hardware configuration object to capture RAID and Fiber Channel information from a target.

RAID and Fiber Channel hardware capture

Capturing RAID and Fiber Channel information requires the use of a vendor-specific environment.

Target inventory for CPU, memory, logical disks, PCI devices, motherboard, and so on, is managed by the OS deployment engine and all information is available immediately if requested. To complete the hardware target inventory with RAID and Fibre Channel information you need the vendor-specific scripting toolkit tools. The hardware capture is done in a similar way to that of the hardware configurations, which means that you need to load the vendor-dependent environment on the target to start the specific capture tool.

The captured hardware information for Fibre Channel and RAID disks can then be seen from the web interface:

Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Creating a hardware environment

To perform hardware configuration tasks on targets, you must have a hardware environment on your OS deployment server.

Before you can create your environment, you must prepare the files on the OS deployment server.

Instructions are provided for preparing the files using scripting toolkits for IBM, Dell, or HP products. It is recommended that you download the latest WinPE 3.x compatible scripting tool environments and use this version. However, the instructions for WinPE 2.x, WinPE 1.x and DOS are also provided.

IBM **IBM ServerGuide Scripting Toolkit WinPE 3.x based**

The following procedure is valid for BIOS targets:

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_utl_sgtkwin_2.30_windows_32-64.zip`.
2. Extract the toolkit into a local directory, for example, `c:\IBM-SGSTK-WinPE3.x`
3. As described in the User's Guide available in `c:\IBM-SGSTK-WinPE3.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, perform the following steps:
   a. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
   b. Install Windows AIK.
   c. Restart your computer.
   d. Expand files `ibm_utl_tsep_2.00_winpe_i386.zip` and `ibm_utl_tsep_2.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE3.x`
   e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
   f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x86.ini` as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

      ```
      SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x86.ini
      ```

      A directory `.\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO` is created and contains the environment tools.

The following procedure is valid for UEFI targets:

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_utl_sgtkwin_2.30_windows_32-64.zip`.
2. Extract the toolkit into a local directory, for example, into `c:\IBM-SGSTK-WinPE3.x`
3. As described in the User's Guide in `c:\IBM-SGSTK-WinPE3.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, you must then do the following:
   a. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows

7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en. and http://www.microsoft.com/download/en/details.aspx?id=5188.

b. Install Windows AIK and supplements by following the Microsoft instructions provided on the download page.
c. Restart your computer.
d. Expand files `ibm_utl_tsep_2.00_winpe_i386.zip` and `ibm_utl_tsep_2.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE3.x`
e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x64.ini` as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

```
SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x64.ini
```

A directory `.\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO` is created and contains the environment tools.

IBM IBM ServerGuide Scripting Toolkit WinPE 2.x based

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_sw_sgtkw_2_1_windows_i386.zip`.
2. Extract the toolkit into a local directory, for example, into `c:\IBM-SGSTK-WinPE2.x`
3. As described in the User's Guide in `c:\IBM-SGSTK-WinPE2.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, you must then do the following:
   a. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008. Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
   b. Install Windows AIK.
   c. Restart your computer.
   d. Expand files `ibm_utl_sep_1.00_winpe_i386.zip` and `ibm_utl_sep_1.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE2.x`
   e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
   f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x86.ini` if you use a 32-bit WinPE2, or `ScenarioINIs\Local\Raid_Config_Only_x64.ini` if you use a 64-bit WinPE2, as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

```
SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x86.ini
```

A directory .\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO is created and contains the environment tools.

IBM

IBM ServerGuide Scripting Toolkit WinPE 1.x based

1. Download the latest ServerGuide scripting toolkit from the IBM website.
2. Extract the toolkit into a local directory, for example, c:\IBM-SGSTK-WinPE1.x.
3. As described in the User's Guide in c:\IBM-SGSTK-WinPE1.x\sgdeploy\SGTKWinPE\Docs\UserGuide.pdf you must then complete the following steps:
   a. Download WinPE 2005.
   b. Run SGTKWinPE.cmd to create a WinPE image with the requested drivers for IBM servers.

IBM

IBM ServerGuide Scripting Toolkit DOS based

1. Download the latest ServerGuide scripting toolkit from the IBM website
2. Extract the toolkit into a local directory, for example, c:\IBM-SGSTK-DOS.

Note: DOS tools are deprecated. They are used only to support some older hardware.

Dell

Dell DTK Scripting Toolkit WinPE 3.x based

The following procedure is valid for BIOS targets:

To set up the WinPE 3.x environment for your Dell servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in English. Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to DTK3.2.1-WINPE-22.exe.
5. Extract the download file. For example, extract the file to the location c:\ Dell-DTK-3.2.1.
6. As described in the Dell User's Guide, in C:\Dell-DTK-3.2.1\Dell\Docs\DTKUG.pdf, you must then complete the following tasks:
   a. Open a command prompt in the directory containing the driver installation batch for WinPE3.x: WINPE3.0_driverinst.bat. For example, the directory, C:\ Dell-DTK-3.2.1\Dell\x32\Drivers\winpe3.x.
   b. Launch the file called WINPE3.0_driverinst.bat *<WINPEPATH>* *<DTKPATH>*, where *<WINPEPATH>* is the destination path to create the directory structure for WinPE 3.0 and *<DTKPATH>* is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called WINPE3.0_driverinst.bat C:\Dell-DTK-3.2.1\

`WinPE3.x_Out_x32 C:\Dell-DTK-3.2.1\Dell\x32\drivers`. Launching this file preinstalls the Dell drivers into `boot.wim`.

The following procedure is valid for UEFI targets:

To set up the WinPE 3.x environment for your Dell servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in English. Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34 &displaylang=en. and http://www.microsoft.com/download/en/details.aspx?id=5188.
2. Install Windows AIK and supplements by following the Microsoft instructions provided on the download page.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to `DTK3.2.1-WINPE-22.exe`.
5. Extract the download file. For example, extract the file to the location `c:\ Dell-DTK-3.2.1`.
6. As described in the Dell User's Guide, in `C:\Dell-DTK-3.2.1\Dell\Docs\DTKUG.pdf`, you must then complete the following tasks:
   a. Open a command prompt in the directory containing the driver installation batch for `x64 WinPE3.x: WINPE3.0_driverinst.bat`. For example, the directory, `C:\ Dell-DTK-3.2.1\Dell\x64\Drivers\winpe3.x`.
   b. Launch the file called `WINPE3.0_driverinst.bat` *`<WINPEPATH>`* *`<DTKPATH>`*, where *<WINPEPATH>* is the destination path to create the directory structure for WinPE 3.0 and *<DTKPATH>* is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called `WINPE3.0_driverinst.bat C:\Dell-DTK-3.2.1\WinPE3.x_Out_x64 C:\Dell-DTK-3.2.1\Dell\x32\drivers`. Launching this file preinstalls the Dell drivers into `boot.wim`.

Dell

Dell DTK Scripting Toolkit WinPE 2.x based

To set up the WinPE2 environment for your Dell servers:

1. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008.Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to `DTK2.6-WINPE-56.exe`.
5. Extract the download file. For example, extract the file to the location `c:\ Dell-DTK-2.6 5`.
6. As described in the Dell User's Guide, in C:\Dell-DTK-2.6\Dell\Toolkit\Docs\DTK25UG.pdf, you must then complete the following tasks:

   a. Open a command prompt in the directory containing the driver installation batch for `WinPE2.x: VPE_driverinst.bat`. For example, the directory, `C:\ Dell-DTK-2.6\Dell\Drivers\winpe2.x`.
   b. Launch the file called `VPE_driverinst.bat` *`<WINPEPATH>`* *`<DTKPATH>`*, where <*WINPEPATH*> is the destination path to create the directory structure for Windows PE 2.0 and <*DTKPATH*> is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called `VPE_driverinst.bat C:\Dell-DTK-2.6\WinPE2.x_Out C:\Dell-DTK-2.6\Dell\drivers`). Launching this file preinstalls the Dell drivers into `winpe.wim`.

7. Copy and rename the customized `C:\Dell-DTK-2.6\WinPE2.x_out\winpe.wim` to `C:\Dell-DTK-2.6\WinPE2.x_Out\ISO\sources\boot.wim`.

Dell

DELL Scripting Toolkit WinPE 1.x based

Note: Windows PE 2005 must be built from a Windows 2003 server for the Dell tools to work.

To set up the WinPE1 environment for your Dell servers:

1. Obtain a Windows PE 2005 file structure.
2. Copy it into a temporary folder, for example, `c:\winpe-dell`
3. The Windows PE 2005 directory structure should contain a directory named `I386` or `MININT`. If it contains a directory named `MININT`, rename it to `I386`.
4. Download the Deployment Toolkit from Dell.
5. Run the executable package to extract the toolkit to the disk of the OS deployment server. In the examples, it is assumed that you have extracted the toolkit into `c:\DELL-DTK`, which implies that you have a folder named `C:\DELL-DTK\Dell\Toolkit`.
6. To install the appropriate drivers for Dell servers in your WinPE image, follow the instructions of the DTK User Guide (*Running Deployment Scripts Using DTK and Windows PE*).

   You must:

   a. Install the drivers with the `driverinst.bat` script
   b. Modify `winpeoem.sif` and `winbom.ini`
   c. Add the RPC DLLs to the Windows PE directory.

      Note: Add the RPC DLLs in `i386\system32` instead of those in the Tools folder.
7. To verify that the drivers are installed, check for the file called `c:\temp\winpedell\i386\system32\racsvc.exe`.

HP

HP SmartStart Scripting Toolkit WinPE 3.x based

To set up the WinPE 3.0environment for your HP servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
2. Install Windows AIK.
3. Restart your computer.

4. Download the latest SmartStart Scripting Toolkit from the HP website: http://h18013.www1.hp.com/products/servers/management/toolkit/. The name of the downloaded file is similar to `SP47335.EXE`.
5. Extract the file into a directory, for example, `C:\HP-TK`.
6. As described in the *HP SmartStart Scripting Toolkit Windows Edition User Guide.pdf* in `C:\HP-TK\SWSetup\SP47335\` and the *Windows Preinstallation Environment User's Guide* (WinPE.chm) contained in Windows AIK, you must then mount the WinPE3.x base image for specific customization. For example, activate extra packages, add drivers, and so on.
   a. From the Windows AIK tools folder, run the command to create WinPE customization directory.

      ```
      C:\Program Files\Windows AIK\Tools\PETools>copype.cmd x86
      C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP
      ```

   b. Mount the base image launching **`Dism`** from the WinPE3.x_HP folder.

      ```
      Dism /Mount-Wim /WimFile:.\winpe.wim /index:1 /MountDir:.\mount
      ```

   c. Install the *neutral* WMI packages in the image.

      ```
      Dism /image:.\mount /Add-Package
      /PackagePath:"C:\Program Files\Windows AIK\Tools\PETools\x86
      \WinPE_FPs\winpe-wmi.cab"
      ```

      Enter the command on one line, although it does not fit on this example.

   d. Install also the language specific WMI package in the image.

      ```
      Dism /image:.\mount /Add-Package
      /PackagePath:"C:\Program Files\Windows AIK\Tools\PETools\x86
      \WinPE_FPs\en-us\winpe-wmi_en-us.cab"
      ```

      Enter the command on one line, although it does not fit on this example.

   e. Add the required drivers (.inf files) to the base image by using the `/Add-Driver` option of the `Dism` command.

      ```
      Dism /image:<mounted image> /Add-Driver /Driver:<driverpath>
      /Recurse
      ```

      Where <*driverpath*> is the location of the .inf files found in the extracted drivers within the hpDrivers folder and `/Recurse` is an option to query all the drivers in subfolders.

      ```
      Dism /image:.\mount /Add-Driver
      /Driver:C:\HP-TK\SWSetup\SP47335\hpDrivers\Winpe30 /Recurse
      ```

      Enter the command on one line, although it does not fit on this example.

   f. Copy the `hpsstkio.sys` Toolkit I/O driver (required for the `conrep` and `rbsureset` utilities) from the HP driver directory to the Windows driver directory. For example:

      ```
      copy C:\HP-TK\SWSetup\SP47335\hpDrivers\Winpe30\system\hpsstkio
      \hpsstkio.sys C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\mount\Windows
      \System32\drivers
      ```

      Enter the command on one line, although it does not fit on this example.

   g. Unmount the customized image to build the customized WinPE.wim:

```
Dism /Unmount-Wim /MountDir:.\mount /Commit
```

7. Copy and rename the customized file

   ```
   C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\WinPE.wim
   ```

   into

   ```
   C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\ISO\sources\boot.wim
   ```

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

HP

HP SmartStart Scripting Toolkit WinPE 2.x based

To set up the WinPE2 environment for your HP servers:

1. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008.Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest SmartStart Scripting Toolkit from the HP website: http://h18013.www1.hp.com/products/servers/management/toolkit/. The name of the downloaded file is similar to `SP38836.EXE`.
5. Extract the file into a directory, for example, `C:\HP-TK`.
6. As described in the *HP SmartStart Scripting Toolkit Windows Edition User Guide.pdf* in `C:\HP-TK\SWSetup\SP38836\` and the *Windows Preinstallation Environment User's Guide* (WinPE.chm) contained in Windows AIK, you must then mount the WinPE2.x base image for specific customization. For example, activate extra packages, add drivers, and so on.
   a. From the Windows AIK tools folder, run the command to create Windows PE customization directory. For example: `C:\Program Files\Windows AIK\Tools\PETools>copype.cmd x86 C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP)`
   b. Mount the base image launching imagex from the WinPE2.x_HP folder. For example, `imagex /mountrw WinPE.wim 1 .\mount`.
   c. Install the WMI packages in the image: `peimg /image=.\mount /install=*WMI*`
   d. Add the required drivers (.inf files) to the base image by using the `peimg /inf` command.

      ```
      peimg /inf=<driverpath> .\mount
      ```

      Where <*driverpath*> is the location of the .inf files found in the extracted drivers within the hpDrivers folder. For example, `peimg /inf=c:\HP-TK\SWSetup\SP38836\hpDrivers\Extr-Drivers\nic\b06nd .\mount`.
   e. Repeat step d. for each additional device driver.
   f. Copy the `hpsstkio.sys` Toolkit I/O driver (required for the `conrep` and `rbsureset` utilities) from the HP driver directory to the Windows driver directory. For example:

      ```
      copy C:\HP-TK\SWSetup\SP38836\hpDrivers\system\hpsstkio\hpsstkio.sys
      C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\mount\Windows\System32\drivers
      ```
   g. When you finish customizing the image, prepare the environment image by using the `peimg /prep` command:

```
peimg /image=.\mount /prep
```

   h. Unmount the customized image to build the customized WinPE.wim:

```
imagex /unmount /commit .\mount
```

7. Copy and rename the customized `C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\WinPE.wim` file into `C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\ISO\sources\boot.wim`.

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

HP

HP SmartStart Scripting Toolkit WinPE 1.x based

The initial setup for the HP SmartStart Scripting Toolkit is similar to the setup of the Dell Hardware Toolkit, because both Toolkits require Windows PE. Some details are not included, but you can read them in the Dell section.

1. Download the Win32 HP SmartStart Scripting Toolkit version of the toolkit on the HP website.
2. Extract it to the disk of the OS deployment server, for example, in `c:\HP-TK`.
3. Create a Windows PE 2005 folder for the HP tools:
   a. Copy a Windows PE file structure to a temporary folder (`c:\winpe_hp`)
   b. Install the HP drivers in the Windows PE directory, as explained in the User Guide for the HP Hardware Toolkit
      1) Run the executable file under hpDrivers
      2) Give the location of the i386 folder of your Windows PE folder

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

To create your environment, perform the following steps:

1. Go to **Server** > **Advanced features** > **Hardware configurations**.
2. Click **New environment** and follow the wizard instructions. You must
   a. Ensure that the web interface extension is running on the computer where Windows AIK and the environment tools have been prepared.
   b. Provide the path of the folder in which the environment tools are located, that is where you have installed the scripting toolkit. For example:

      IBM

      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO (BIOS)`
      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO (UEFI)`

      Dell

      - `C:\Dell-DTK-3.2.1\Dell\x32 (BIOS)`
      - `C:\Dell-DTK-3.2.1\Dell\x64 (UEFI)`

      HP `C:\HP-TK\SWSetup\SP47335`

   c. Provide the path of the folder in which the environment material is located, that is the WinPE files. For example:

      IBM

- `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO (BIOS)`
- `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO (UEFI)`

Dell

- `C:\Dell-DTK-3.2.1\WinPE3.x_Out_x32\ISO (BIOS)`
- `C:\Dell-DTK-3.2.1\WinPE3.x_Out_x64\ISO (UEFI)`

HP `C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\ISO`

To view the created environment go to **Server** > **Advanced features** > **Hardware configurations**. Alternatively, you can also go to **Server** > **OS deployment** > **Software modules**. To view the created environment, look under a specific environment folder.

Now, you can create hardware configurations using this environment.

Creating a hardware configuration object

A wizard allows you to easily create hardware configuration objects.

Before you can create a hardware configuration object, you must have created the environments needed to later perform the hardware configuration tasks.

1. Go to **Server** > **Advanced features** > **Hardware configurations**.
2. Click **New hardware config.**.
3. Select the kind of hardware configuration that you want to create.
4. Provide at least one target model and environment pair on which the hardware configuration can apply.
5. For BIOS update, BIOS settings, or Hardware custom configuration the specific files or set of files can be downloaded from the specific vendor sites.
6. Follow the wizard instructions.

To view or edit a hardware configuration, select the hardware configuration and select **View configuration details** in the contextual menu. In the **Hardware configuration details**, use the **Edit** buttons to update the different sections.

Creating a hardware capture configuration

A wizard allows you to easily create hardware capture configuration in a way similar to that for hardware configurations.

Before you can create a hardware capture configuration, you must have created the environments needed to later run the hardware capture.

- If you do not yet have a hardware capture configuration, perform the following steps:
  1. Go to **Server** > **Advanced features** > **Hardware configurations**.
  2. Click **New hardware config.**
  3. Select **Hardware discovery**.
  4. Provide at least one target model and environment pair on which the hardware capture can apply.
  5. Follow the instructions of the wizard.
- If you already have a hardware capture configuration, you can add target model and environment pairs, as follows:

1. go to **Server** > **Advanced features** > **Hardware configurations**.
2. Select **Hardware discovery**.
3. Double-click **Hardware capture configuration**.
4. Under **Hardware environment matching**, click **Edit**.
5. Click **Add a new line** and select the model and environment values
6. Repeat step 5 for each pair to be added.
7. Click **OK**.
8. Click **Back** to return to **Server** > **OS deployment** > **Hardware configurations**.

To view or edit the hardware capture configuration, go to **Server** > **Advanced features** > **Hardware configurations**. Select **Hardware discovery**, and double-click the hardware capture configuration. In the **Hardware configuration details** page, click **Edit** to update the different sections.

You can now capture RAID or Fiber Channel information.

Capturing hardware information using templates

When you capture hardware information with templates, this capture is done every time the template is used.

Capturing hardware information with templates requires an additional reboot to boot the specific hardware configuration environment (WinPE, DOS,...) and launch the specific scripting toolkit tools.

Note: You can not capture hardware information from a target started with a network boot media.

Capturing hardware information with templates always tries to capture both RAID and Fiber Channel. To run the capture:

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select **Idle Layout** or **Deployment Schemes**, depending on which state you want to perform the hardware capture. If you select **Deployment Schemes**, the discovery is performed at deployment time.
3. Double-click the chosen template to view its details.
4. Click **Edit** on **General settings**.
5. Under **Perform inventory on:**, select **RAID**. Select this option in the deployment scheme only if you are creating a hardware configuration for the hardware capture. In this way you avoid a failure at any target PXE boot.
6. Click **OK**.

Capturing hardware information once

When you want to capture hardware information only once for a target, or a group of targets, you do this with a specific tool.

Capturing hardware information requires an additional reboot to boot the specific hardware configuration environment (WinPE, DOS,...) and launch the specific scripting toolkit tools.

Note: You can not capture hardware information from a target started with a network boot media.

1. Go to **Server** > **OS deployment** > **Target Monitor**.
2. Select a target or a group of targets.
3. Select **Additional features** from the contextual menu.
4. Double-click the chosen template to view its details.
5. Select **Capture hardware parameters** and click **Next**.
6. Select **Raid capture**, **Fiber channel capture**, or both, and click **Next**.
7. Follow the instructions of the wizard.

When captured, the RAID and Fiber channel information can be viewed. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. On this page look under the **Inventory** tab.

Task templates for Windows operating systems

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout
The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout
The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout
The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning

profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout
The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Customizing a screen layout

You can customize the screen layout of a target.

To customize a screen layout:

1. Select the layout that you want to customize in the right pane of the **Task Templates** page of the web interface page.

   Note: An actual layout must be selected and not a layout folder (left pane)
2. At the bottom of the page, the screen layout is shown in reduced size. Click **Customize GUI** to open the screen layout editor.
3. The editor is composed of a left column, containing instructions, a *What-You-See-Is-What-You-Get* (WYSIWYG) view of the screen being edited and a bottom banner with action buttons.
4. Click on the action buttons or directly on the items that you want to modify to see their editable properties displayed in the left column. Make the wanted changes and then click **Save** to keep your new screen design. Return to the **Task Templates** page by clicking **Back**.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard. Go to **Server** > **OS deployment** > **Task templates**. and click **New deployment scheme**.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section. Some advanced deployment scheme features are available only in this mode and not through the wizard.

3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   The following parameters apply for simple one-time deployments and for redeployment operations.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, so the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters
   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

   Edit parameters for targets that are not yet in the database
   to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

   Never edit parameters
   to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

   You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

   No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

   Yes, display a warning
   if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

   Yes, abort the deployment
   if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

   Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, because Tivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of targets to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed target, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, `/` on UNIX operating systems or `C:/` on Windows operating systems. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: Multicast is available only if:

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of targets that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast
to deploy targets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast

- **without explicit synchronization**

  to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.
- **with synchronization on all targets**: to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.
- **with the following synchronization parameters:**

  to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

In addition to that, you can also configure two additional parameters:

- `Multicast block size (MB)` with default value 16
- `Multicast TOC block size (KB)` with default value 32

During a deployment configured to use multicast, the server splits the system profiles in blocks that are sent to all the targets in the same multicast group. Before sending each block, the server sends a packet, called ToC, that describes the content of a block. The parameter `Multicast block size` is the maximum size of a block of data. The second parameter `Multicast TOC block size` is the maximum size of the ToC.

These parameters can be configured to improve the performance of the multicast deployment. If you have a high speed network rated at 1 GB and a profile of 15 or 20 GB, it is recommended to change both the values to at least 64.

For Windows Vista and later, and Linux platforms, you can decide to use a network share on Windows, or Linux HTTP, to download the files to the targets , rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP provides a shorter installation time. For Linux deployments, you can also allow the use of the Linux Deployment Engine for 64-bit cloned profiles using PXE.

To use a network share on Windows or Linux HTTP:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter in the **Network share module** section. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module).

On Linux only, to use the Linux Deployment Engine:

- Select **Use Linux Deployment Engine x86-64 if applicable** to deploy a Linux cloned 64-bit profile using PXE, for driver injection purposes. This option is ignored in all other cases.

On-site deployment

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment option

Indicate if you want to keep the deployment image in a protected partition and the size of this partition. These options are valid only to configure the deployment scheme for redeployment. More information is available in deploy/tosd_redeplscheme.dita.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation

This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

This parameter, located in the **General settings** section, is set to **No** by default. Setting this parameter to **Yes** unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user. Review the section on network boot scenario of the deployment process topic.

Unbind software module at the end

This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is relevant for Windows Vista/2008/7/2012 deployments only.

Send mail at end

This option can be set only if a *sendmail* tunnel has been created.

Send mail to

This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network

This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button

This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Creating media for offline deployment of Windows operating systems

You can create deployment media such as CDs, DVDs, or USB drives to install machines without connecting them to the OS deployment server.

You can use this kind of deployment when there is no connection or connection to the OS deployment server is very slow.

Some typical situations are small branch offices with slow links and no local deployment server, isolated computers with no connection to an internal network, laptop users currently away from LAN or connected using a modem.

If the data you want to use does not fit on a single CD or DVD, use a USB drive.

You can also create offline Win PE-based deployment media in kernel-free mode. You select the deployment engine type and the architecture to use, as well as System Profiles and any software module you want to include in the deployment media. If you have more than one WinPE deployment engine, you can select the target models and the matching engines for these models are included. The engines are cached locally on the target.

Note:

- You must create the deployment media from an OS deployment server or a web interface extension installed on a computer with the same byte order (little endian or big endian) as the one on which you want to use the deployment media.
- To deploy Windows system profiles on Hyper-V, make sure that the boot order indicates the hard drive before the CD-ROM or USB drive.
- Redeployment is not available when deploying from a deployment media.

Creating a kernel-based OS deployment USB drive with the wizard

Tivoli Provisioning Manager for Images can automatically generate deployment USB drives that replay the deployment process for a given system profile or for any kind of software modules available.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must be formatted as FAT32 or NTFS.

Note: SuSE Linux Enterprise Desktop cloning is not supported on USB drive deployments.

Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The deployment USB drive is self-contained and can be used instead of a CD or DVD to provision a target entirely offline, without using the OS deployment server. These deployment USB drives can also be used to deploy computers without a PXE-compliant network adapter.

To create OS deployment USB drives:

1. Perform one of the following operations:

- Go to **Server** > **OS deployment** > **Task templates**.
- Go to **Server** > **OS deployment** > **System Profiles**.
- Go to **Server** > **OS deployment** > **Software modules**.

2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment USB key** to start the USB key wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. (*Optional*) Change settings for targets running the USB key that you are creating.

   Included objects

   When selecting objects to be included, be aware that:

   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile and exactly one deployment scheme must be included in your image.
   - The software application order is automatically included.
7. If your USB key has already been used as a deployment media, you might choose to keep a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB key.
8. Plug your USB key into a machine running the web interface extensionand specify its address.
9. Choose the drive matching your USB key.
10. Click **Finish** to close the wizard.

Use the USB drive to deploy a given system profile or any kind of software module.

Creating a kernel-based OS deployment USB drive with command lines

You can create an OS deployment USB drive that Tivoli Provisioning Manager for Images can use when a target cannot boot from the network.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must have boot capabilities and a FAT32 or NTFS filesystem. The drive must be already formatted; existing files on the partition are not deleted. USB keys already filled with a bootable operating system might not work.

Note: Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The command line must be used only when the web interface is either inappropriate or unavailable.

Use this command line:

- On Windows operating systems:

```
rbagent.exe -s <OSD_server_ip_address>:<OSD_server_password>
rad-usbget <drive>
keepshared|delshared preferwpe|prefermcp nodes
```

Where:

OSD_server_ip_address
Is the IP address of the OS deployment server.

OSD_server_password
Is the password for the administrative user (typically admin) on your OS deployment server.

drive Is a drive letter of the Windows target where you run the rbagent command. The rad-usbget command adds requested files to the FAT32 or NTFS partition and makes it bootable. The drive must be already formatted. Existing files on the partition are not deleted.

keepshared
Keeps a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB drive.

delshared
Deletes a shared repository of previous data.

preferwpe|prefermcp
Defines if the Linux deployment engine environment or WinPE is loaded from the USB drive, when a target boots from this USB drive, without accessing the network. Only when Linux deployment engine or WinPE is running, does the target connect to the network and try to contact an OS deployment server. If you deploy only Linux, specify prefermcp to skip WinPE. You can specify preferwpe only if there is a WinPE deployment engine on the OS deployment server.

nodes Defines the deployment settings with a space-separated list of objects. Specify at least DEPLSET:Default for the deployment schema, and PROFILE:SystemID for the system profile.

You can now boot the target using the OS deployment USB drive instead of the network card. To use the PXE emulation USB key, insert the USB key into the drive and restart the target. If your machine does not boot from the USB key, check the BIOS boot list to see if your optical drive is included in the boot sequence and is listed before the hard disk. Most machines also allow you to select the temporary boot device without changing the boot sequence in BIOS.

Creating a kernel-based OS deployment CD or DVD

Tivoli Provisioning Manager for Images can automatically generate deployment CDs and DVDs that replay the deployment process for a given system profile or for any kind of software modules available. You can use this feature to create OS deployment CDs and DVDs that can be easily sent through the Internet or by e-mail, to refresh a computer back to its initial working state after installation.

The CD/DVD deployment occurs in kernel mode. Microsoft tools are used to build the CD/DVD. By specifying the target models, the product automatically determines which deployment engine to use and the drivers corresponding to the specified target models are added to the CD/DVD. These CDs and DVDs can also be used to deploy computers without PXE compliant network adapter. The creation of DVDs and media spanning is supported. These media can be protected using an activation code preventing unauthorized personnel from using it.

To create OS deployment CD and DVD:

1. Perform one of the following operations:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.
   - Go to **Server** > **OS deployment** > **Software modules**.
2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment CD or DVD** to start the CD and DVD wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. Follow the wizard instructions to create an ISO image.

   Included objects

   When selecting objects to be included in the ISO image, be aware that:
   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile and exactly one deployment scheme must be included in you image.
   - The software application order is automatically included.

   Hardware options

   In the hardware options settings some boot options can be customized. By default the options are unchecked but some special cases can require changes. In particular, if the CD or DVD is to be used on a USB drive or as a secondary drive, it might be necessary to specify the option **use BIOS for CD or DVD ROM access**. When this option is selected, on some hardware it might also be necessary to select **disable enhanced disk access** (for IDE CD or DVD) or **disable USB** (for USB CD or DVD) to ensure that Tivoli Provisioning Manager for Images use of other IDE or USB devices does not interfere with the BIOS access to the CD or DVD. In addition, deploying from the second CD or DVD drive of a target only works if you can ensure that subsequent boots keeps booting on the same CD or DVD drive.

   Security issues

   For security issues, you might want to protect deployment from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment to proceed.

You might also want to hide the content of the ISO image that contains sensitive information such as product keys. To do this, select **Hide the content of CD or DVD** in the CD or DVD Wizard. If you then try to access files in your ISO image, you see the content as `CDROM_content_hidden`.

Size of the ISO file

The wizard allows you to choose the size of the ISO images.

a. Enter the maximum size in the field displayed.

b. Click **Next** and the wizard starts to precompute the ISO file size.

The wizard displays the results for the number of disk images and the size required. You then have the option to:

- Download it directly from the server.
- Use the web interface extension
- Generate it on the server itself in the import directory.
- Generate it on another computer running the web interface extension

Note:

- When creating the ISO files, all objects of type *single file to copy*, *image headers*, and *WIM images* (which includes Windows Vista/2008/7/2012 unattended setup profiles), are put on the first CD or DVD. Therefore, the first ISO file might grow larger than the requested spanning size if the total size of the files to be put on the first ISO requires it.

  For example, if you try to create an OS deployment DVD containing both Windows Vista/2008/7/2012 unattended setup profiles, both profiles must be contained on the first ISO, but their total size is larger than 4 GB. Therefore, the ISO cannot be burned into a single layer DVD. In this case, either use a double layer DVD, or transfer the ISO without burning it.
- When deciding where to generate the ISO image, be aware that:
  - If the estimated size is bigger that 2 GB, do not use the link to download directly from the server, because of limitations of web browsers. An exception to this rule is Mozilla Firefox on Linux, which can extract files as large as 4 GB or more.
  - Because of file system limitations, do not extract files bigger than 4 GB on FAT32 partitions.

Use a CD creation tool to burn the ISO image onto disks.

Note: Vista 2008 Windows 7 Windows Vista/2008/7/2012 unattended setup profiles contain at least one file larger than 1 GB which cannot be split. Therefore, ISO files containing Windows Vista/2008/7/2012 unattended setup profiles must be burned on a DVD.

If you encounter problems when deploying from this CD or DVD on a virtual machine, make sure that the CD drive comes after the hard disk in the boot order.

Setting up an activation code

For security issues, you might want to protect deployment or booting from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment or the network boot to proceed.

To prevent being asked several times for the activation code during deployment:

- The deployment scheme included on your deployment CD must have the network setting **Use 'BIOS fall back MBR' to start PXE** set to **No**.
- The boot order of your target must be set to hard disk first and you must boot on the CD manually the first time.
- To set up an activation code for the first time, when creating the deployment CD:
  1. Select **Include activation code protection** in the deployment media wizard.
  2. Enter and confirm the chosen password. You must remember this password if you want to obtain other activation codes for this CD.
  3. Set a password expiration date under **Valid until**.
- To obtain a new activation code, for example, if you must use the CD after the current activation code expiration date:
  1. Click **Generate Media** on the Profiles page to start the deployment media wizard.
  2. Select **Generate a new activation code**.
  3. Click **Next** and follow the wizard instructions to obtain your new activation code. You must remember the password given when creating the first activation code for this CD.

The wizard provides you with the generated activation code that you need when using the CD.

Creating a standard WinPE-based CD/DVD or USB for deployment in kernel-free mode

For Windows operating systems without a network connection, you can create an offline CD/DVD or USB for deployments in kernel-free mode.

Before creating the Windows PE-based offline deployment CD, DVD or USB drive , ensure that you have configured your WinPE deployment engines to match your target models and to contain the critical drivers for the specific target hardware.

Create a CD/DVD or USB drive for offline deployments in kernel-free mode:

1. Go to the **Task templates**, the **System Profiles**, or the **Software modules** page.
2. Click **Generate media** at the bottom of the page.
3. Select **Create a standard Windows PE-based media** and click **Next**.
4. Select both the deployment engine type and the architecture to use based on the operating system you want to install. If you choose the **Select the best available (Auto)** option and both 32-bitWinPE 3.x and WinPE 4.x are available, the system selects WinPE 4.x. You can override the default engine architecture and select the 64-bit architecture.

   Note: You must select the same WinPE architecture of the target operating system in the USB, CD, or DVD media. WinPE is used for the first boot from the selected media, during the setup phase, and also as the final action from cache.
5. If you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.

6. Select the **Inject all available drivers** check box, if you are creating your media based on a model which is not known to Tivoli Provisioning Manager for OS Deployment. Using this option, all compatible drivers are added.
7. Select at least one System Profile that you want to include in the offline deployment media. Optionally select one or more software modules, and click **Next**.
8. Choose the desired media: Select **CD/DVD** or **USB key** .

   Depending on your selection:

   CD/DVD

   If your target startup sequence is first CD or DVD and second hard disk, make sure that **CD/DVD will boot at user request only** is selected to boot from the CD/DVD only after user interaction. This is because the deployment flow must first start with a boot from the WinPE deployment engine available in the CD/DVD while successive boots must be performed from the hard disk where the WinPE deployment engine was cached. For this reason, the first time that the target boots, you must press any key to boot from the CD/DVD and start the deployment. Successive unattended reboots fall back to hard disk until the deployment is completed.

   Click **Next** .

   USB key

   Click **Next** .
9. Specify the machine where the Web Interface extension (rbagent) is installed. It can be local or on another computer.

   CD/DVD

   This is the computer on which the offline deployment WinPE-based ISO image is created

   USB Key

   This is the computer on which your USB key must be plugged in. All deployment files are downloaded to this drive.

   Note: If the Web Interface extension is running on a machine with windows XP or Windows 2003, you must prepare the USB drive by formatting it in advance as FAT32 or NTFS. Click **Next** .
10. For the selected media:

    CD/DVD

    Specify the path and file name of the ISO file that will contain the deployment engine.

    USB The available USB drives on the selected machine are displayed. Choose the drive matching the key where the deployment files must be downloaded.
11. After a few minutes the CD/DVD or the USB media is created. Click **Finish** to close the wizard.

The generated ISO file or USB contents can be deployed on the target.

Important: If your deployment media is an USB drive, the hard disk device must be configured before the USB drive in the target boot sequence to successfully deploy the operating system. Force the boot from the USB device to start the offline deployment.

Deploying Windows operating systems

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

The deployment process

In Tivoli Provisioning Manager for Images, a deployment is made up of several steps that are automatically run in sequence without user interaction:

1. Hardware configurations are optionally deployed, for example, to create RAID volumes.
2. Partitions are created on the hard disk, and then formatted according to information contained in the system profile.
3. All deployment objects (system profiles, partition files, and software modules) are downloaded to a temporary storage location on the hard disk.
4. Operating system files are written in the hard disk partitions, creating a bootable operating system with files and applications configured by database bindings between the *target* and *software modules*.
5. Target-specific configuration, such as the *host name* or the *product key* are gathered from the database to create a textual configuration file used by the system preparation tool.
6. The operating system is started, allowing Sysprep to configure the operating system according to information stored in the Tivoli Provisioning Manager for Images database.
7. Additional software is optionally installed, if it must be installed after the operating system.
8. The temporary storage location is cleaned. Installation files are removed.
9. Tivoli Provisioning Manager for Images takes control again when Sysprep has completed and rebooted the target, and displays a message indicating the status of the deployment.

When the deployment is complete, the operating system is installed and ready to be used by the user defined for this target in the database.

Network boot scenarios

Depending on the number of OS configurations bound to a specific target, a target behaves differently when it boots on the network:

- If no OS configuration is bound to the target (for example, when a target starts for the first time and has not been configured), a special screen is displayed that asks the administrator to configure an OS configuration binding for this target on the OS deployment server. Deployment is not possible until an OS configuration is bound to the target.
- If one or more OS configurations is bound to this target, but no deployment has been scheduled on the server, a screen is displayed with a list of all the OS configurations bound to the target. Clicking on an item in the list starts an interactive deployment for the selected OS configuration, using either the **Default** deployment scheme (if no deployment scheme has been configured for this target), or the deployment scheme used during the last deployment.
- If one or more OS configurations are bound to this target, and a deployment has been scheduled on the server for a specific OS configuration, the target immediately starts the deployment without requiring any user intervention.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

Note:

- To deploy a Windows operating system, you must have a WinPE 3.x deployment engine stored on your OS deployment server.
- During the deployment, do not edit the WinPE deployment engine that you are using.
- The system profile you are deploying cannot contain partitions labelled with letter *P*, *Q*, *X*, or *W*. These letters are reserved.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Restrictions on user privileges

On Windows Vista/2008/7/2012 cloning system profiles, it is not possible to give a user administrator privileges if the user name existed in the reference target without these administrator privileges. Trying to do so, either from the **Target details** page or from the **OS configuration details** page, results in a failed deployment.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

Windows To deploy any Windows system profile, you must have a WinPE deployment engine stored on your OS deployment server.

Vista 2008 Windows 7 For provisioning Windows Vista, Windows 7, and Windows 2008 profiles, use a WinPE 3.x deployment engine.

2012 Windows 8 For provisioning Windows 8 and Windows Server 2012 profiles, use a WinPE 4.x deployment engine.

Vista 2008 Windows 7 Here are the requirements to deploy Windows Vista/2008/7/2012/8.

- To deploy an unattended setup profile for Windows Vista/2008/7/2012 32-bit, the minimal size of the hard drive of your target is about 10 GB.
- To deploy an unattended setup profile for Windows Vista/2008/7/2012 64-bit, the minimal size of the hard drive of your target is about 20 GB.
- If you have a Volume Licence edition, the product key field in the system profile details must be empty for the deployment to succeed.

When you deploy a cloning system profile, you may encounter driver and/or disk mapping issues if the hardware of the reference target of the cloning system profile is significantly different from the hardware of the deployment target. This is particularly true for 64-bit operating systems. In this case, you should try to transform your cloning system profile into a universal image.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. If you do not choose this option, select the type or types of deployment you want to perform. You can install additional software only if you deploy an operating system.
   a. If you have selected **Perform hardware configuration tasks**, indicate which hardware configurations you want to deploy.
5. Select **Simple deployment** and click **Next**

6. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Three options are available in the deployment wizard to deploy targets without physically interacting with the computers.

   - **Try to wake up targets currently powered off using WOL** asks the Target Monitor to send IBM Wake on LAN packets to wake up targets. Waking up targets only works on carefully designed modern computers. A target can only be woken up if its network adapter and its system board support Wake on LAN packets, and if the network adapter has been shut down properly. If the network adapter is not in the appropriate power state, Wake on LAN packets will not wake the computer up. This is not specific to Tivoli Provisioning Manager for Images, but is rather a general limitation of the Wake on LAN technology.
   - **Try to wake up targets using management interface** asks the Target Monitor to contact the targets and send a reboot request. If you are running the web interface extension that uses specific arguments starting with **rad-**, you might not be able to reboot targets remotely. They must be rebooted manually. You need the web interface extension running with the correct privileges to run a remote boot.
   - **Try to reboot targets running theweb interface extension** asks the Target Monitor to contact the targets if they are running under Windows and send a reboot request. If you are not running Windows, you cannot reboot targets remotely. They must be rebooted manually. If you are running Windows, you need the web interface extension running with the correct privileges to run a remote boot.

   If you have not selected one of these options or if they do not work, and if the target you are trying to deploy is not powered on, turn it on now and make it start on the network.
7. If your system profile uses the **driver specific bindings mode**, a check is performed to warn you of potential driver issues. If your system profile uses the **regular software binding rules**, this check cannot be performed and a warning message is displayed. If you want to switch from one binding mode to another, you must do it on the system profile itself, on the **Profile details** page.

For a Windows cloning deployment, the target goes through the following stages:

1. **Prepare one partition** Tivoli Provisioning Manager for Images creates partitions on the hard-disk according to the information stored in the system profile associated with the OS configuration being deployed.
2. **Install Operating System files** Tivoli Provisioning Manager for Images downloads deployment files on the hard-disk and installs the operating system.
3. **Generate Windows Sysprep configuration file** Tivoli Provisioning Manager for Images creates the files needed by Sysprep.
4. **Windows Sysprep Mini-Setup** Tivoli Provisioning Manager for Images runs Sysprep Mini-Setup.
5. **Install additional software** Tivoli Provisioning Manager for Images installs the various software modules according to their application order, handling multiple reboots if required.
6. **Complete Windows post-OS configuration** Tivoli Provisioning Manager for Images finalizes operating system settings not configured by Sysprep.
7. **Cleanup deployment data** Tivoli Provisioning Manager for Images deletes deployment files.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

If you want to test the operating system deployed, you must first ensure that the target is not configured to start on the network, or you will get a menu allowing you to run the deployment again.

Windows Vista/2008/7/2012 offline servicing

Offline servicing allows the OS deployment server to patch Windows Vista/2008/7/2012/8 image with HotFixes and language packs before the deployed operating system needs to be connected to the network, thus preventing the risk of contracting viruses before the operating system is fully functional and is patched with security updates.

Offline servicing also enables you to use language packs with versions of Windows Vista other than Enterprise or Ultimate.

Offline servicing is automatically performed on aWindows Vista/2008/7/2012/8 deployment when

- A HotFix (.msu) or a language pack is bound to the OS configuration.
- The `unattend.xml` parameter file contains a `<servicing>` tag.

To perform offline servicing, you must have a WinPE 3.x or WinPE 4.x deployment engine on your server.

Deploying a hardware configuration

A wizard allows you to effortlessly deploy hardware configurations.

To start a hardware configuration deployment you must first have at least a hardware configuration environment and a hardware configuration.

Note: You can not deploy a hardware configuration from a target started with a network boot media.

1. Select a single target or multiple targets on the Target Monitor page. To do this go to **Server** > **OS deployment** > **Target Monitor**. To select multiple targets or deployment, select an **administrative group** , a **custom list**, a **subnet**, or click on **individual target** names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the second screen of the deployment wizard, you must select at least **Perform hardware configuration** tasks and if you want to proceed with the Operating System/Software deployments you can also select another option.
4. Select one or several hardware configuration(s) you want to apply on target. RAID Configurations, BIOS Settings, BIOS Updates or Hardware custom configurations are classified in a matching folders.
5. Follow the deployment Wizard as it is described in the chapter Deploying depending on the options you chose above.

Every configuration you selected will automatically use the appropriate environment and only be applied if the model is matching the target.

Redeploying

This function is a special deployment scheme that gives you the ability to rapidly restore an image to a computer from a hidden partition on the computers hard-disk.

During the original image deployment to the computer, Tivoli Provisioning Manager for Images creates a hidden partition on the hard-disk of the target computer. When it has finished deploying the master image on the computer, it stores a reference image into the hidden partition. It is possible to store one or more reference images into a hidden partition on the computer.

Note: Before running a deployment task on a machine with a redeployment partition, ensure you remove the hard disk partition content.

Each time the system is booted, either off the hard-disk or using network boot, Tivoli Provisioning Manager for Images intercepts the boot process of the computer and presents a customizable menu of possible actions. Those actions are:

- Boot the system off the current image on the hard-disk.
- Do a quick cleanup of the currently deployed image against the reference image and restore the image from the hidden partition.
- Do a format and full restore of the reference image from the hidden partition. Using this function, it is possible to effectively have a fresh image deployment every day for the optimum performance of a system.
- Choose and deploy another configuration available on the hidden partition. This option takes as long as the format and restore option.

Note: Redeployment is not available when deploying from deployment media or when booting in kernel-free mode.

The purpose of redeployment

A computer generally works the best and the fastest on the day that it is installed. At that time, the system is completely clean, free of any undesirable processor-consuming gadgets, and all programs are configured for their optimal use by the system administrator. The purpose of redeployment is to ensure that the system is reset to this optimal state at every boot (or at some fixed interval).

There are three categories of systems that experience the most visible need for the redeployment technology:

Public computers
such as schools, universities, and Internet cafes, where users cannot be relied on to preserve the computer integrity, because the computer is not their own

Critical systems
such as banks, insurance companies, and industrial plants, where the company cannot afford to risk computers being reconfigured or infected by malicious software

Embedded systems
such as ticket machines, airport information systems and ATMs, that must be quickly rebuilt to their original OS configuration, without using a specific infrastructure

Because redeployment often occurs at the user's desk, it is necessary to find a solution that is quick, easy to use, does not require any significant infrastructure, and does not affect the work process of other users. This rules out standard deployment tools, because they impose a significant load on the network and affect other users' ability to perform their tasks.

Note: The redeployment feature is not intended to be used on virtual machines. On virtual machines, you should leverage the snapshot feature of your hypervisor rather than use the redeployment feature.

The redeployment process

Redeployment involves several steps, including creating a reference image of the target, and saving it as a redeployment partition.

Redeployment steps

Tivoli Provisioning Manager for Images addresses the challenge of redeployment with the following steps:

- At the end of a deployment, Tivoli Provisioning Manager for Images creates a reference image of the target, and saves it into a protected redeployment partition (invisible to the user and to the operating system itself). This increases deployment time by roughly 10% compared to a simple deployment, as most of the files are already present as file archives on the disk at that time.
- Every time a target starts, Tivoli Provisioning Manager for Images hooks the boot process before the operating system starts (using PXE or a special Master Boot Record).
- If configured to do so, Tivoli Provisioning Manager for Images authenticates the user of the target against the server database to restrict the use or the maintenance of the target to authorized persons only.
- If configured to do so, Tivoli Provisioning Manager for Images offers the choice of several OS configurations available on the target (multiboot), and of several levels of "cleaning".
- Using the reference image saved during deployment, Tivoli Provisioning Manager for Images resynchronizes the hard-disk content to its reference state. This typically takes only a few seconds, but can take up to a few minutes if everything on the hard disk has been deleted.

Offline redeployment limitations

Offline redeployment behaves slightly differently from online redeployment as the OS deployment server cannot be contacted for information. These limitations are removed after the target contacts the OS deployment server again. For example, interrupted tasks are not automatically resumed and changes to the partition scheme cannot be recovered.

Moreover, authentication with offline redeployment does not work. A message warns the user.

Note: If you plan to use redeployment with multiple OS configurations offline, make sure that all the preloaded OS configurations have exactly the same partition layout (number and size), because Tivoli Provisioning Manager for Images cannot create new partitions offline or to resize existing partitions offline. Failure to do so prevents you from redeploying offline some of the preloaded OS configurations.

Redeployment with multiple operating systems

You can preload up to three operating systems on a target, with a menu allowing the user to select which operating system to start.

Scenario

You want to provision the computers of a classroom with two different operating systems (for example, Windows 2008 R2, by unattended setup, and RHEL by cloning). When entering the classroom, the student must choose between the two operating systems. For security reasons, you want to make sure that the operating system which is started is always in a clean state. You also want the selected operating system to install and start quickly.

Principles

To achieve this, you must install each operating system in its own partition, save the OS configurations in a protected partition. Before you start an operating system, you do a rapid verification of the operating system partition with the information in the protected partition.

Requirements

For you multiple operating systems to cohabit in a single target and to be able to start them individually, you must follow these guidelines strictly:

- The hard disk of the targets must be large enough to contain the three operating systems and the protected partition.
- You must create a separate system profile for each operating system.
- All the profiles must have the same number of partitions, in the same format.
- Each operating system must be in a distinct partition, and all other partitions must be empty during the system profile creation.
- Each operating system must be in a primary partition, and there is a maximum of three primary partitions.
- In the system profiles, partition numbers cannot be modified.
- An offline refresh does not update the partition table.

Before you start, select the target machine to be used to create the system profiles needed. For example, this procedure can be executed with a hard disk of 40 GB, and 12 GB are assigned to each operating system.

Procedure

1. 1. Create your Windows 2008 R2 unattended system profile.
   a. From the web interface, create a new unattended system profile with the profile wizard, following the instructions.
      1) Create one primary and bootable NTFS partition for Windows 2008 R2 and determine the right size for this partition.
   b. Customize your OS configuration.
      1) Set the administrator name in the configuration.
      2) Optionally, bind software modules.
   c. Deploy a target with this profile.
2. Create your RHEL cloning system profile.
   a. Start the target where you deployed the Windows 2008 R2 with the RHEL installation CD.

   b. In the Linux installer, in the partitioning steps, select the option “Create custom layout, and modify the layout as follows:
      1) Leave unchanged the first NTFS partition.
      2) Create one primary partition (EXT3) for / (partition 2) and determine the right size for this partition.
      3) Create another primary partition of 1 GB for Linux swap.
      4) Install the GRUB loader.
   c. Install RHEL in partition 2.
   d. Customize your installation.
   e. After the installation, from the web interface, clone your target to create a new linux system profile.
   f. After the clone, check the partition layout retrieved by the OSD. In particular check and correct if needed the following settings:
      1) For the NTFS partition:
         a) Edit the size to the right value.
         b) Set the bootable flag to no.
      2) For the Linux EXT3 partition:
         a) Set the bootable flag to yes.
      3) For the Linux swap partition:
         a) Set the size to the right value (1 GB)
3. Update the two OS configurations.
   a. Edit the partition scheme for each OS configuration so that partitions have the same size on each OS configuration.
      1) For the unattended Windows 2008 R2 system profile you must create two Linux partitions, one EXT3 primary not bootable for / and one Linux swap primary of 1 GB.
   b. Use the best size found for each operating system.
   c. Set the options Must be deployed and Must be redeployed so that only the partition containing the operating system (and also the swap partition in the Linux case) it is actually deployed or redeployed for each system profile.
4. Test each system profile. Each operating system installs in the correct partition, without impacting other partitions.
5. Create a specific deployment scheme for this redeployment.
   a. Export the two system profiles into a RAD file. This step in only used to determine the partition size, as explained at step 3.c.
   b. With the deployment scheme wizard, create a new deployment scheme enabling redeployment.
   c. For **Protected redeployment partition size**, give 200% of the size of the RAD file you have just created.
6. Preload the system profiles on your targets.
   a. Select the targets in the web interface.
   b. Select **Deploy now** in the contextual menu.
   c. Select **Redeployment preload** in the deployment wizard.
   d. Select the deployment scheme you have just created.
   e. Select the two OS configurations that you have prepared.
   f. Optionally, select additional software modules.
   g. Click **Customize GUI** if you want to customize the boot menu appearing on the target.

Scenario

You want to perform a multi-boot installation with three operating systems which have all unattended deployments.

Procedure

1. Create each unattended profile, and do not duplicate existing profiles. At creation time, define in the profile wizard the partitions where you must deploy the different operating systems. The partition must be the same for each profile. For example, if you deploy the first profile in the first partition, the second profile in the second partition and the third profile in the third partition you must create each profile as follows:
   - Create the first profile by specifying in the profile wizard three partitions and checking the first partition as the one on which the operating system should be installed.
   - Create the second profile by specifying in the profile wizard three partitions (same partition layout as in the first profile) and checking the second partition as the one on which the operating system should be installed.
   - Create the third profile by specifying in the profile wizard three partitions (same partition layout as in the first profile) and checking the third partition as the one on which the operating system should be installed.

   In this way, the first profile has the first partition marked as bootable, the second profile has the second partition marked as bootable, and the third profile has the third partition marked as bootable.
2. Edit each profile partition layout and set the **Must be deployed** and **Must be redeployed** options to **No** on the partitions where the operating system is not installed. Set the **Must be deployed** and **Must be redeployed** options to **Yes** on the partitions where the operating system is installed. The result should be the following in the three profiles:

Table 5. Partition options in the first profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	Yes	No	No
Must be redeployed	Yes	No	No

Table 6. Partition options in the second profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	No	Yes	No
Must be redeployed	No	Yes	No

Table 7. Partition options in the third profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	No	No	Yes
Must be redeployed	No	No	Yes

3. Export the profiles into a single rad file and set the size of the redeployment partition in the scheme as 200% of the rad size at least.

4. Deploy the profile configurations into a single deploy now operation on the target specifying the redeployment. Using the profile wizard, customize the GUI to have three options with the possibility to boot on each deployed operating system.

Configuring a deployment scheme for redeployment

Redeployment is a feature that affects *how* the target is being preinstalled, not *what* is in the deployed OS configuration. Redeployment is enabled by customizing a deployment scheme.

Because redeployment is basically the replay of a standard deployment operation, you must first configure a regular deployment process, and try it on a test computer. When you have performed these two stages, follow the instructions provided to turn your one-time deployment OS configuration into a redeployment OS configuration.

To customize a deployment scheme for redeployment, you can

- Create a new deployment scheme with the deployment Scheme Wizard
- Modify an existing deployment scheme with the deployment Wizard
- Edit the parameters of an existing deployment scheme manually

The following steps are based on the first and second options, which are very similar.

1. Follow the first alternative to create a completely new scheme, and the second alternative to modify an existing scheme with the wizard:
   - Go to the **Task templates** page and click **New deployment scheme**. This launches the deployment Scheme Wizard, which guides you through the customization of deployment parameters.
   - Go to the **Task templates** page. Select a deployment scheme, and click **Edit parameters using a wizard**.
2. Follow the instructions of the wizard in the same way as for a regular deployment, until you reach the panel called **On-site deployment features**.
3. Select **Enable support for quick redeployment of the same OS configuration** and click **Next**.
4. On the next panel, **Redeployment option**, select **Yes, keep** IBM Tivoli Provisioning Manager for Images **images in a protected partition**. Optionally modify the space that you want to allocate to this special partition, and click **Next**.

   Note:

   a. The protected partition size must be at least as large as the total size of all system and software images to be deployed on the computer, because it retains all these images. If you are unsure of the space required, start with approximately 800 MB for a Windows 2000 configuration, 1500 MB for a Windows XP configuration, or 1500 MB for a Linux configuration. If you want a more precise number, check the image sizes reported in a deployment log, and round up the total to accommodate the miscellaneous structures used for redeployment.
   b. The space that you allocate to the redeployment partition is subtracted from the hard-disk total capacity detected by Windows or Linux. The user cannot detect, access, or delete this protected area from the operating system disk manager. It is not simply a hidden partition, but a hardware-protected area,

as defined in ATA-5 specification. If necessary, you can recover this space by running another deployment operation.

5. Click **Finish** to complete the customization process and obtain a deployment scheme ready for redeployment.

Edit the parameters manually:

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select a deployment scheme
3. Click **View deployment parameters**
4. Click **Edit** in the section header in which you want to modify parameters.

Preloading for redeployment

Before you can redeploy a target, you must preload one or several OS configurations.

For a successful redeployment, targets must not **Boot on hard-disk if idle**. Make sure this target parameter is not selected for the targets you want to redeploy.

After you have created an appropriate redeployment scheme, you can begin the preload of the OS configurations of your choice on the target. This operation must be initiated using the Target Monitor page of the web interface.

1. Select the targets to deploy and select **Deploy now** from the contextual menu to start the deployment wizard.
2. Select **Redeployment preload** and click **Next**.
3. Follow the instructions of the deployment wizard.

   Note:

   a. When you select a deployment scheme, only those configured for redeployment are displayed. If you do not have any scheme ready for redeployment, a warning message appears.

   b. Preloading more than one OS configuration is supported, but increases the preload time.

The preload automatically starts when the targets boot, just like with regular one-time deployments. The process goes through the same steps, with one exception. When Sysprep or `LinPrep` has completed and after all software modules have been installed, an image of the fully configured target is stored on the redeployment partition. If you have selected multiple OS configurations, the process repeats for all OS configurations in turn, until all redeployment images are ready.

Customizing the redeployment menu

You can customize the menu entries that you see in the user interface when starting a target in redeployment mode. Each OS configuration can define one or more menu entries, and the complete menu is the union of all entries defined by all available OS configurations.

After having selected **Redeployment preload** in the deployment wizard and selected the deployment objects:

1. Click **Customize GUI** in the deployment wizard. This opens the menu customization interface which is divided into three parts:
   - A left column with instructions on how to modify the menus and editable fields

- A bottom banner with action buttons
- A view of the target screen as it will appear

2. Click **New menu item**.
3. Modify the captions and actions.
4. You can select one of the following actions:
   - **Format and restore**
   - **Quick restore**
   - **Boot on OS**
5. If you want to protect a specific menu item from unauthorized users, you can set up a global password or user authentication for that user by selecting an appropriate value under **Authentication**. To make full use of this feature, you must first have defined authentication domains in the **Server parameters**. Three authentication formalisms are available

   Authenticate locally on RAD *group*
   uses the local user database to authenticate a user. The optional *group* parameter can be used to restrict the verification to a specific group of users. This type of domain is supported by both Windows NT and UNIX versions of the OS deployment server.

   Authenticate on NT server *server:group*
   forwards authentication requests to the NT server specified by the mandatory parameter *server*. The optional parameter *group* can be used to restrict the verification to a specific group of users. This type of domain is supported by the Windows implementation of the OS deployment server only.

   Authenticate on Radius server *ipaddr:secret*
   forwards authentication requests to the Radius-compliant device specified by the parameter *ipaddr*. The value of the parameter *secret* is used as the secret for the Radius communication, and must match the secret stored in the configuration of the Radius device for the protocol to work.

   Note: Authentication with redeployment does not work if the target is offline (the target has no network connection and boots from the hard disk). A message warns the user. If you plan to redeploy offline, use a global password rather than user authentication.
6. Click **Save** and then **Close** to exit this window.

Formatting hard disk and restoring files:

With this option, your partitions are always reformatted and all the files restored before you boot into the operating system.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network). If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

On the target, select the OS configuration to be restored.

After an OS configuration has been selected, Tivoli Provisioning Manager for Images completely format the disk and then restore all files. The default behavior is to:

1. Format the disk partitions as specified in the system profile.
2. Restore all the files from the hidden partition.
3. Boot on the selected operating system.

Using quick redeployment:

This option is the typical way to use redeployment. A fast verification of partitions and files is run and, fixes are performed if needed before the target boots into the operating system.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network). If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

On the target, select the OS configuration to be restored.

After an OS configuration has been selected, Tivoli Provisioning Manager for Images automatically restores it as quickly as possible. The default behavior (which typically takes only a few seconds to run) is to:

1. Verify that the disk partitions match the wanted system profile, and fix them if needed.
2. Verify that all partitions have the appropriate file content, and fix them if needed.
3. Boot on the selected operating system.

Booting on the installed operating system:

This option allows you to boot on the currently installed operating system, without any verification. It is fast, but it does not prevent operating system corruption.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network).

If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

On the target, select the option that allows you to boot on the operating system.

The target boots directly in the installed operating system, without any disk partition or file verification.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for Windows operating systems

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State
Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     - **No**
     - **Whatever the notification is**
     - **If deployment is successfully completed**
     - **If deployment failed**
  5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.

- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Bindings created during deployment

The Target Monitor creates a binding between the OS configuration chosen for the deployment and the targets being deployed. This binding is added into the database and can be later removed using the Target Monitor.

Because at least one configuration binding now exists, targets that have been deployed no longer show the locked screen. They show a boot menu with a list of the OS configurations that are bound to the target. This allows the target user to manually restart the deployment of an already deployed OS configuration by clicking on the corresponding line in the menu.

What you can do

You can remove, add, or modify OS configurations and software bindings using the Target Monitor.

Chapter 3. Provisioning Linux operating systems on x86 and x86-64 targets

This section provides information on how to work with the product to deploy Linux operating systems.

System profiles for Linux operating systems

A system profile is the partition layout and list of files to deploy an operating system, either by unattended setup or by cloning, from a reference target or from a reference image file.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- When deploying Red Hat Enterprise Linux (RHEL) Version 6, both 32- and 64-bit, on IBM servers with more than one disk, you must force the disk mapping of the first disk in your profile to `/dev/sda`.
- To deploy a Red Hat system profile with LVM partitions on a XEN guest, modify the default device mapping of your system profile, and force the first disk to `/dev/hdb`.
- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.
- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.
- You cannot deploy Linux profile with an LVM root partition if you use deployment media.

Creating system profiles

There are distinct types of system profiles. The profile wizard guides you through the creation of system profiles for each type.

Creating an unattended setup system profile for Linux operating systems

You can install operating systems using standard installation processes in unattended mode. Unattended setup simplifies the task of preparing computers for the native mode of operation of disk cloning.

During deployment of a Linux unattended setup profile, /swap is used as temporary cache partition. It should be at least 2 GB to hold all the necessary files.

Note: If you are deploying Linux on machines with two disks, ensure you modified the ks.cfg file of the **OS configuration details** page with one of the following statements:

```
bootloader --driveorder=sdb,sda
```

or

```
bootloader --driveorder=hdb,hda
```

depending on the disk naming system of the machines.

To create a new system profile:

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select your operating system from the list and click **Next**.
5. Follow the instruction of the profile wizard.

When your first unattended installation profile is created, you can use it to deploy targets. Then you can create a cloning-mode system profile, because unattended installation profiles have a longer deployment time than cloning-mode system profiles. You can use your unattended installation profile to prepare the computer that you refer to when creating your first cloning-mode system profile.

Creating an unattended setup system profile for Red Hat Linux V4.9:

To install Red Hat Enterprise Linux 4.9 you must upgrade a Red Hat Enterprise Linux 4.8 installation using the **up2date** command, because you cannot create a Red Hat Enterprise Linux 4.9 system profile.

Modify the sources file contained in the /etc/sysconfig/rhn/ directory of Red Hat 4.8, to specify from where the **up2date** command downloads the upgrade files of a Red Hat 4.8 64-bit. In this example the local FTP server is myftpserver:

```
yum rhel-os ftp://mylogin:mypassword@myftpserver/redhat/
yum/4/es/os/x86_64/
yum rhel-updates ftp://mylogin:mypassword@myftpserver/redhat/
yum/4/es/updates/x86_64/
yum rhel-extras ftp://mylogin:mypassword@myftpserver/redhat/
yum/4/es/extras/x86_64/
```

Customize the sources file according to your environment. For more information about **up2date**, see the Red Hat documentation.

1. Create a system profile of Red Hat 4.8 using the system profile creation wizard
2. Create a Linux software module that runs the **up2date** command at the end of the deployment:

a. In the Software Module wizard, select: **A Linux software module**
b. Select **A custom action on the target computer**
c. Select **A configuration change to perform on the target computer (a command to execute...)**
d. Select **Copy a single text file**
e. Choose the computer where the `sources` file is located.
f. Select your `sources` file
g. At the end of the wizard, choose to apply your software module **When the OS is installed**. Specify `/etc/sysconfig/rhn/` as the target destination path.
h. Edit the created software module by specifying in the **Command line** option: `rpm --import /usr/share/rhn/RPM-GPG-KEY; up2date --update <=2h` This command imports the default GPG key needed by **`up2date`**, and then runs the **`up2date`** command with a timeout of 2 hours. You can increase the timeout if needed.

3. Deploy your Red Hat 4.8 system profile with the software module to run the **`up2date`** command.

At the end of the deployment, you have installed Red Hat Linux 4.9.

Creating an unattended setup system profile for Red Hat Linux V5.x:

To install Red Hat Enterprise Linux Client 5.x using the Workstation option, after installing a new OS Deployment server or upgrading it to the current build, you must create a new unattended system profile for Red Hat Enterprise Linux Client 5.x. If the system profile was created with a previous build, it cannot be modified to add the Workstation option.

To enable the Workstation option, you must have a specific product key provided by Red Hat.

The Workstation option can be added by providing this product key during the profile wizard, while creating a new unattended system profile for Red Hat Enterprise Linux Client 5.x, or editing its OS configuration details after creating the system profile.

If you want to add it during the new system profile wizard, in the OS configurations panel, you can find a new check box that, if selected, allows you to enter the key. The key must be in the following format:

```
xxxx-xxxx-xxxx-xxxx
```

You can also avoid selecting the option during the profile creation and add it later, editing the Fixed Unix-specific properties in the Unix tab of the OS configuration details of your Red Hat Enterprise Linux Client 5.x unattended system profile.

Note: No check about the correctness of the key is performed during the system profile creation, but the Workstation option will not be added during the installation, if the key is wrong.

When you provide the correct key, the software groups that you selected during the Red Hat Enterprise Linux Client 5.x unattended system profile creation, will include also the Workstation packages.

If you create the `RH_PROFILE_INCLUDE_ALL_RPM` file, as described in the information center, to include all packages from the installation media sources, only the Linux

HTTP download mode can be used, which is the default download mode. You can verify if the Linux HTTP download mode is selected by editing the deployment scheme that you want to use for the Red Hat Enterprise Linux Client 5.x unattended system profile deployment.

Creating a cloning-mode system profile for Linux operating systems

To obtain a cloning-mode system profile from a reference target you must first prepare the reference target.

Note: As of version 7.1, LVM2 is supported for cloning. During the cloning process, the physical volume under LVM2 is moved to an extended partition.

The product supports only one volume group per disk and a volume group cannot span over two disks.

Preparing the reference target:

To create a cloning-mode system profile, you must first create the reference OS configuration, the *system profile*, that you want to deploy.

You must perform this task on the reference target and not on the OS deployment server.

The OS deployment server does not perform cleanup on the reference target. You must delete useless files and services before creating a new image:

- Delete the temporary internet cache.
- Delete your temporary directories and files.
- Disconnect your network drives and remote printers.
- Empty the recycle bin.
- Delete partitions that use a file system not supported by the product, or reformat the partitions.

  Note: On RHEL 6, Ext4 is supported, except for the `/boot` partition which must be in Ext3.

Preparing a Linux system profile:

When you prepare a Linux system profile, consider the space for the temporary cache partition and the bootloader.

Ensure that the partitioning scheme provides enough space for the temporary cache partition during deployment. For Linux cloning, `/boot` is used as temporary cache partition. It must be large enough to hold image file headers and software modules. The recommended size is 256 to 512 MB, unless you have very large software modules and must augment this size. If you do not want to change the `/boot` partition of your reference computer, you can edit the size of the `/boot` partition directly in the cloned system profile.

Tivoli Provisioning Manager for Images supports only Grand Unified Bootloader (GNU GRUB). You can install GRUB on the bootsector of the Linux `/boot` partition or on the root partition. If you plan to use redeployment, you must install GRUB in the boot sector of the Linux `/boot` partition. To start your system with GRUB, ensure that you have a standard MBR on the disk, with the boot partition flagged as bootable.

You do not have to run a system preparation tool for deploying Linux using Tivoli Provisioning Manager for Images. Tivoli Provisioning Manager for Images automatically installs and runs its own system preparation tool, LinPrep.

The Xen virtualization package part of RHEL5 is not supported. Remove the Xen package from your reference computer before you clone it.

NTFS and exFAT partitions are not supported in Linux system profiles. Use FAT 32 partitions instead.

Cloning the reference computer:

After you have prepared your reference computer, you can create your system profile from the profile wizard in the web interface.

- The reference target must be ready to PXE-boot into the OS deployment server.
- Shut down the the reference target before starting the cloning process. If it is not shut down, ensure that the web interface extension is not running.

1. Go to **Server** > **OS deployment** > **System Profiles**.
2. Click **New profile**.
3. Select **Cloning from a reference machine** and click **Next**.
4. Complete the profile wizard.

Creating a universal system profile for Linux operating systems

When creating a software module, do not enter a hardware model because a universal system profile must be deployable on several types of hardware. If you entered a model name in the Profile Wizard, you can delete it when you edit the first set of parameters of the Profile details.

To deploy your universal system profile with a type of hard disk different from that of your reference target (for example, going from a parallel hard disk to an SCSI or an AHCI disk), the system handles hardware changes by rebuilding the initial ramdisk, or initrd, during deployment. The rebuilding of initrd is available for 32-bit cloned images only.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles

To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Browsing partition files

You can browse partition images stored on your server.

1. Go to **Server** > **OS deployment** > **System profiles**. Double-click on a profile to view the details.
2. In the **Original partition layout** section, click **Browse image of primary partition 1**.
3. You can expand or update the whole partition or a part of it.
   - To expand the whole or part of the partition:
     a. Right-click the folder you want and select **Expand on local disk**.
     b. Choose the computer where you want to expand and store the files contained in the selected partition.
     c. Specify the destination folder where to extract the partition files.
     d. Follow the instructions of the image wizard to expand the partition.

        Note: You must expand the partition to an empty directory. If you select a folder that is not empty the extraction fails.
   - To update the whole or part of the partition:
     a. Right-click the folder you want and select **Update from local disk**.
     b. Specify the source folder of the OS deployment server where your updated data are located.
     c. Specify the destination folder where to extract the partition files.
     d. Follow the instructions of the image wizard to update the partition.

On the **Partition image explorer** page, you can create a new directory by selecting **Add new directory** in the contextual menu. You can also modify or add files by selecting **Upload file** in the contextual menu.

Note: File upload is limited to 16 MB.

Changing the partition layout

You can update partition layout to resize partitions, assign mount points, or change the file system.

If you change the partition layout in system profiles, you might render the profile unusable. It is recommended not to change the partition layout in system profiles, unless you know that the changes you want to make have no side effect.

Do not transform a primary partition into a logical partition.

Note: Changing the partition layout from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose either one or the other entry point, and then perform all your changes from that entry point.

By editing the partition layout, you can:

- Add or delete partitions.

Note: Adding or deleting partitions can lead to OS configuration problems. Use this feature carefully. To provide a clear description to your profile, use the **Comment** field.

- Resize a partition by dragging sliders, or by assigning it an absolute or relative size.
- Change the file system of a partition.
- Assign a mount point to the partition.

Note: When in the disk partitioning, both fixed sizes and percentage sizes are included, the exact reproduction is not guaranteed.

1. Click **Edit partition layout** on either the **Profile details** page or the **OS configuration details** page, **Disks** tab.
2.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

        Linux In a Linux profile, assign a mount point for the new partition. To be valid, this mount point must reference an existing directory in the main image. Starting from Fix Pack 3, the Linux profiles with the root partition as LVM are supported. In this case, you must ensure that the HTTP mode is selected in the deployment scheme when deploying the profile. With the root partition as LVM, you cannot perform the deployment using the media.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.
3.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

        Windows In a Windows profile, the operating system deployed using a system profile must be installed on the C: drive. Other drive letters are not allowed for the bootable partition.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.
4.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

Modified partitions are aligned on megabytes rather than on cylinders. The following warning message might appear in the logs and can be safely discarded: `Warning: partition x does not end at a cylinder boundary`. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

If you want to use the same system profile with two different partition schemes, you can also duplicate a system profile by right-clicking the profile name and selecting **Duplicate profile**. The copy shares the same image files, but can have a different partition layout.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.
3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

   Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/cxdx`, where *x* is the disk number. Edit your device mappings accordingly.
4. Click **OK**.
5. Repeat step 2 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies unique.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Editing OS configuration parameters in the web interface for Linux

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Editing a Linux parameter file

Note: Starting with version 7.1.1, information about partitions in the custom configuration file is not normally taken into account.

SUSE For partitioning information in the custom configuration file to be taken into account, and to replace any information in the default file, these conditions must be fulfilled:

- The version of the product must be 7.1.1.3 or higher
- The deployment must be performed by HTTP
- The system profile must be of type *unattended setup*
- The operating system being deployed must be SuSE

Information in **Common networking info** is overwritten by the information in the custom configuration file. However, information in the **Advanced network settings** is not overwritten because it is applied post-configuration.

1. On **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**:
   - Red Hat To edit the file, click **Edit custom 'ks.cfg'**.

     Note: If you are deploying Linux on machines with two disks, add one of the following statements to the `ks.cfg` file:

     ```
     bootloader --driveorder=sdb,sda
     ```

     or

     ```
     bootloader --driveorder=hdb,hda
     ```

     depending on the disk naming system of the machines.

- SUSE To edit the file, click **Edit custom 'autoinst.xml'**.

  You can use the following sections in your file:
  - `<files>`
  - `<groups>`
  - `<users>`
  - `<signature-handling>`
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). The resulting configuration is the union of the values in the custom and default files, with the following restrictions:

- The result of conflicting values between the custom and default files is undefined.
- Partition information in the custom file is taken into account only for SuSE unattended setup by HTTP, in which case only the information in the custom file is taken into account.
- Advanced network settings are always applied, because they are performed at a later stage.

SUSE Here is a short example of a `autoinst.xml` file which adds a new user during setup.

```
<profile xmlns="http://www.suse.com/1.0/yast2ns"
         xmlns:config="http://www.suse.com/1.0/configns">
 <users config:type="list">
   <user>
     <username>jdoe</username>
     <user_password>tOpsEcreT</user_password>
     <encrypted config:type="boolean">false</encrypted>
     <forename>John</forename>
     <surname>Doe</surname>
   </user>
 </users>
</profile>
```

Do not omit the `xmlns` and `xmlns:config` attributes of the `profile` tag.

Troubleshooting:

If the OS configurations in the deployed operating system are not what you expected, you must examine the parameter files. They are the result of the merge between the custom file and the default file created.

Red Hat To troubleshoot OS configuration parameters after a failed deployment, complete the following procedure:

- Without rebooting the target:
  1. Type `Alt+F2` on the target. This opens a shell.
  2. In the opened shell, view the file `/tmp/anaconda.log`.
- Look for `ks.cfg` at the root of the partition labeled `rembo`. The file contains the information merged from the custom and the default files.

SUSE To troubleshoot OS configuration parameters after a failed deployment, there are the following options:

- Without rebooting the target:
  1. Type `Alt+F2` on the target. This opens a shell.
  2. In the opened shell, view the file`/var/log/YaST2/y2log` .
- You must look for `autoinst.xml` at the root of the partition labelled `rembo`. The file contains the information merged from the custom and the default files.

Software modules for Linux operating systems

Software modules are images other than system profiles that can be created to address various needs.

Tivoli Provisioning Manager for Images is based on imaging technology. As administrator, you create images of components that you want to see on every target, and the automated deployment merges and restores these images on each target, automatically, when needed.

Tivoli Provisioning Manager for Images can handle most scenarios for software deployment and post-installation configuration.

Types of software modules

There are many types of software modules. Depending on the type of package and installation files, the wizard guides you through the different steps to achieve your software module with minimal effort. The types of software package supported by the wizard are listed in this section.

- **A Linux application installation, using RPM**
- **A custom action on the target computer**. This includes OS configuration changes such as commands to be run, and copying sets of files on the target.

Creating software modules

There are distinct types of software modules which vary according to the operating system being deployed. The software wizard guides you through the creation of software modules for each type.

Creating software modules with RPM for Linux operating systems

Using RPM for Linux software installations.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **Linux** and click **Next**.
4. Select **A Linux application installation, using RPM** and click **Next**.
5. Complete the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.

- The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
- A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
- The path to where the installation files are restored on the target. This path is relative to the system root partition.
- An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

  For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=`*`x`*`h`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

  ```
  install /sPB /rs /rps /l <=1h
  ```
- The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a custom action software module for Linux operating systems

Software modules can also contain custom actions to be performed on the target.

They are divided into:

- An OS configuration change to perform on the target
- A set of files to copy on the target

Configuration changes are further subdivided. Depending on the operating system, you can:

- Copy a single text file
- Run a single command file, this can be a batch file or a vb script file.
- Boot a virtual floppy disk

In the OS configuration change wizard screen, you can select **Activate keyword substitutions**. If you use this option, you can specify which keywords must be substituted in the software module details.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select the operating system and click **Next**.
4. Select **A custom action on the target** and click **Next**.

5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The full path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

     For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=xh`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

     ```
     install /sPB /rs /rps /l <=1h
     ```
   - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

Repeating custom actions:

Some commands must be run every time the target boots during a deployment.

This is typically the case if you want to repeatedly connect a network share. This connection is destroyed when rebooting. You can therefore create a single software module with a `netuse` command to set the network share and set this software module to run once after each reboot, starting at a specific reboot.

This option is available for executing a single command.

1. Create your software module.
2. Double-click on the software module name in the **Software components** page to obtain the **Software details** page
3. Click **Edit** in the title of the **Package information** section.
4. Select the installation stage at which the software module must be applied first.
5. Select **Run at each software pass until end of deployment** and click **OK**.

Creating a software group

Simplify the management of your software modules by grouping them into containers called *software groups*.

A *software group* is a collection of software modules that behaves as a standard software module.

The advantage of software groups is to manipulate only one object instead of several software modules when they should all behave in the same way. For example, you can select a whole software group for deployment, create a binding rule for it, or change its software application order, instead of doing it for each software module individually.

The elements of a software group are individual software modules. You cannot nest software groups within software groups.

A software module can belong to several software groups simultaneously.

To create a software group:

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **A software group** and click **Next**.
4. Select all the software modules that you want to include in your software group and click **Next**.
5. Follow the remaining instructions of the wizard to create your software group.

You can now create binding rules for your software group, modify its application order, export it to a RAD file, or use it in a deployment, as if it were a standard software module.

You can also edit the software group, for example to add or remove software modules.

Editing software modules

You can edit the basic parameters of a software module, upload new files into your software module, and update drivers.

1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
2. From **Software details** page, use the links and buttons. Links vary according to the type of software module. Not all the links listed are necessary available.
   - To edit the base parameters of a software module, click **Edit** at the top of the **Software module information** section.
   - To update files or add new files into the software module, click **Edit software module files**, or a link with a similar name, and select **Upload file** from the contextual menu.

     Note: File upload is limited to 16 MB.
   - For software groups, to add or remove software modules:
     a. Click **Edit** at the top of the **Software group contents** section.
     b. Select the software modules that you want to add.
     c. Deselect the software modules that you want to remove.
     d. Click **OK**.

Keeping command lines confidential

When you use command lines in your software modules, their call and their output are stored in deployment logs. In some circumstances, for example when the command line includes a password or a product key, it might be necessary to keep the information contained in the command line confidential. Three levels of confidentiality are available.

No confidentiality
The command line is visible in the web interface and on the target during the installation, its call is logged, and its output is also logged.

The command line call is not logged
The command line is visible in the web interface, and its output is logged, but the command line call, containing the whole command line string with all parameters, is visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by one exclamation mark (`!`).

The command line call and output are not logged
The command line is visible in the web interface, but its call and output are visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by two exclamation marks (`!!`).

To keep command lines confidential:

- Enter the appropriate number of exclamation points in front of the command in the Software Wizard when first creating the software module.
- Edit the software module information
  1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
  2. Click **Edit** in the Software module information banner.
  3. Update the command line with the appropriate number of exclamation points.
  4. Click **OK**.

Keyword substitution

You can usefully use keyword which act as variables and are substituted with their values during deployments. Keywords can either refer database values or server specific values, given by the user.

Syntax

Variable substitution expressions follow the syntax given here. They start with the character { and end on the same line with }. Words between these two characters are interpreted by using one of the following schemes:

- *{$expr$}* the expression is replaced with the string resulting of the evaluation of `expr`.
- *{/expr/ab}* the expression is replaced with the string resulting of the evaluation of `expr`, but each occurrence of the character `"a"` is replaced by the character `"b"` (character-based substitution).

- *{=expr=test content=this is a test}* the text "this is a test" is included in the destination file only if the string resulting of the evaluation of expr is equal to the text "test content".
- *{!expr!test content!this is a test}* the text "this is a test" is included in the destination file only if the string resulting of the evaluation of expr is not equal to the text "test content".

Note: If a variable does not exist (for example, it contains a typing error or it is not described in server.ini) but it is used in a command, its value is supposed to be empty which can result in deployment errors.

Database keywords

Within an expression, database records can be referred to. Within a record, each field can be accessed using the standard C notation (record.fieldname). The exhaustive list of these fields can be obtained from the database records, with the following correspondences between variable and database record names:

Table 8. Records for free-text conditions

Variable record name	Database record name
Disk	DiskInventory
DMI	DMIInventory
Order	BOM
User	UserProfile
System	SystemProfile
PCI	PCIInventory

Below are a few examples of available fields:

- Order.IP: a string, the target IP address, such as 192.168.1.2
- Order.MAC: a string, the target MAC address, such as 00:01:02:03:04:05
- Order.SN: a string, the target Serial Number, such as CH12345678
- Order.Model: a string, the computer model name, such as e-Vectra
- User.UserCateg0: a string, without any restriction, such astechnicians
- DMI.Vendor: a string, the vendor name, such as Hewlett-Packard
- DMI.Product: a string, same as Order.Model
- DMI.ProcModel: a string, the processor model
- Disk[0].Type: a string, the disk 0 drive type, such as ATAPI
- Disk[0].Media: a string, the disk 0 media type, such as Disk or CD
- Disk[0].DiskSize: a number, the physical size of the disk (if detected)
- PCI[0].VendorID: a string, the hexadecimal vendor ID of the device
- PCI[0].DeviceID: a string, the hexadecimal device ID of the device

For disks and PCI devices, you can use the function sizeof (sizeof(Disk) and sizeof(PCI)) to discover the number of devices present. You can then use indexes to access these devices.

As an example for keyword substitution, if BomID has OrgName Rembo SaRL, RemboServer 192.168.168.16, and IP 192.168.168.32 for value 1, the following text

```
BomID:{$Order.BomID$}
OrgName:{$User.OrgName$}/{$StrToLower(User.OrgName)$}
RemboServer:{$Order.RemboServer$}
IP:{$Order.IP$}
```

gives the following results after keywords are substituted (note the use of a Rembo-C function within the expression to be substituted):

```
BomID:1
OrgName:Rembo SaRl/rembo sarl
RemboServer:192.168.168.16
IP:192.168.168.32
```

Server specific keywords

If you want to set up server specific keywords, which are defined exclusively by the user and per server, you must edit `Files/global/rad/server.ini`.

Start the file with `[Custom]` and add a line per keyword, in the format **keyword=value**, where `keyword` is a word of your choice and value the `value` you want to give it.

To use the keyword in a command, type `Server.keyword` and activate keyword substitution when creating the software module.

Note: `server.ini` is not replicated between servers. If you use multiple servers, you must edit `server.ini` on each server.

Customizing the software page

You can view the software modules in a tree viewer or in a list viewer. The list viewer allows you to customize the visible information.

You must have created at least one software module, otherwise there is nothing to view.

To customize the visible information

1. Go to **Server** > **OS deployment** > **Software modules**. Then click **List view**.
2. From the list view, you can
   - Drag the column separator in the column heading to resize the column.
   - Click on the triangular arrow to the left of the column name to sort the software modules by column criteria.
   - Click on the arrow on the right of the column name and select an option to filter the information. Filtering on several columns is cumulative.
3. For more options, right click anywhere to open the contextual menu and select **Arrange columns**.
   - Select the columns you want to see and clear the others.
   - Click on the minus or plus icons to decrease or increase the size of a column.
   - Select a column and use the up and down arrows to move the column relatively to the others.

   Click **OK** to save your changes. The updated version of the list view is visible in the **Software modules** page.

To return to the tree view, click **Tree view**. You can also access the details of the software modules by double-clicking on a software module name, from either view.

OS configuration and software bindings

OS configuration bindings determine which configurations are available to a target when booting the target on the network, while software bindings correspond to the list of software modules currently assigned to the target.

OS configuration and software bindings are created when:

- The Target Monitor has been used to manually modify OS configuration and software bindings for the target
- A deployment has been started with the Target Monitor. In this case, an OS configuration binding is added for the corresponding OS configuration.
- Automatic binding rules are configured in the **Details** page of OS configurations or software modules. Some of these rules have matching values for the specified criteria. These bindings cannot be modified, except by modifying the rules.

With the Target Monitor, you can browse, remove or add OS configuration and software bindings to any target present in the database. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Binding software modules and OS configurations to targets

Bindings link software modules and OS configurations to targets to enable automatic deployment. When binding to targets, you explicitly provide the list of software modules and OS configurations to bind to your target.

To explicitly bind a software module or a OS configurations to a target, there are two methods:

- From the **Target Monitor** page
- From the **Target details** page

If you want to bind software modules or OS configurations to a group of targets, you must do it through the Target Monitor.

From the Target Monitor:

1. Select a target or a group of targets
2. Select **Bind software** or **Bind OS configurations** from the contextual menu
3. Select the items to bind from the popup window
4. Click **OK**

From the Target details page:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.
2. Go to the **Bindings** panel.
3. Click **Edit** in the relevant section to add explicit bindings for OS configurations and software modules.
4. Select the items for which you want to add explicit bindings.
5. Click **OK**

You can also clear items to remove their explicit bindings. To remove a binding by rule, you must modify the rule.

Binding software modules to a deployment scheme

Software modules can be bound to deployment schemes.

Take a company with offices in three locations: New York, Quebec City, and Mexico City. In each of these locations, the company has people in human resources, sales, logistics, and product development. For the sake of simplicity, consider further that all the employees use either one of two types of computers: a desktop, or a notebook. All desktop computers are identical (with the same network card, system board, disks, and so on) and the same applies for all notebooks.

In this scenario, the company needs two profiles, one with the image for notebooks and one with the image for desktop computers. Three configurations per profile (six in total) are necessary to integrate the different parameters of the different locations, in particular language and time zone information. Finally, schemes are set according to the employees' department, with software modules specific to the different departments bound directly to the deployment schemes.

1. Go to **Server** > **OS deployment** > **Task templates** Select the **Deployment Schemes** folder. Double-click on a deployment scheme to view its details.
2. Click **Edit** on the **Software bindings** section of the page to open the dialog to bind software modules to schemes.
3. Select which software modules you want to bind to your deployment scheme, in addition to software modules that can have been bound to targets.
4. *(Optional)* If you want to use only the software checked in the window when deploying with this scheme, select the **Discard all other software binding rules** check box.

Automatic binding rules

Automatic binding rules are used to create bindings between OS configurations and targets, or software modules and targets, without having to specifically bind a OS configuration or a software module on each target.

Rules are created in OS configurations and software modules to determine which targets are automatically bound to the OS configuration or software module.

Rules are made of criteria and values. If a target has a matching value for all criteria in the rule, the OS configuration or software module will be bound to that target. The binding will be displayed with the mention **by rule** in the OS configuration panel of the target properties for targets that match the criteria. For example, if the criteria is the model name, and the value is `Optiplex`, targets with a model name starting with `Optiplex` will be bound to the object where the rule has been defined.

Automatic binding rules are defined in Tivoli Provisioning Manager for OS deployment at the bottom of the **OS configuration details** or **Software details** page.

To create a new binding rule, click **New rule** located at the bottom of the Web interface:

1. The dialog displayed to create a new binding rule is different depending on whether you are adding a rule to an OS configuration or to a software module. When adding a binding rule to a software module, you can set values for the following criteria:
   - A deployment scheme
   - A system profile
   - A current OS configuration
   - Administrative group

- One of the system-definable and user-definable fields of the database (only used if you have customized the database)
- An operating system type, such as Windows 2000
- An operating system version, such as SP2
- An operating system language
- An operating system architecture, such as x86-32
- A computer model name
- A BIOS version
- A PCI device
- A base board
- MultiChassi
- HAL Type
- A free-text condition in `Rembo-C;` syntax

For example, to create a binding based on the operating system type between a software module and targets, you must create a new rule, click **OS type**, and select the operating system version that you want to limit this software module to.

2. When adding a binding rule to an OS configuration, you can set a condition on the deployment scheme, and on the computer model name. The next ten fields are only used if you have customized your database and want to match specific user categories.
3. Finally, you can enter a free-text condition following the `Rembo-C;` syntax. They must only be used by advanced users.

The conditions determine the applicability of the rule and evaluate to `true` or `false`. A condition must be formed using the variables also used for keyword substitutions in software modules, combined with Java-like logical operators, listed by order of priority in the table:

Table 9. Logical operators for free-text conditions

Operator	Meaning
<	smaller than
<=	smaller than or equal to
=>	greater than or equal to
>	greater than
==	equal to
!=	not equal to
&&	AND operator
||	OR operator

For example, a typical condition can be:

```
Disk[0].DiskSize > 10*1024*1024
```

Note: If a condition cannot be evaluated, it is considered to have the value `false`.

Scheduling the application of software modules for Linux operating systems

Tivoli Provisioning Manager for Images provides a wide flexibility in the specification of a deployment task. As several software modules can be deployed in conjunction with a system profile, you can schedule when they must be applied.

Tivoli Provisioning Manager for Images provides a wide flexibility in the specification of a deployment task. As several software modules can be deployed in conjunction with a system profile, you can schedule when they must be applied.

Typical application locations for software modules include:

- Sysprep and unattended setup processes are automatically run during the OS installation phase, if required.
- For other software: when the OS is installed or after additional reboots depending on the software module needs.

Software modules are not ordered within an installation stage. If you want a software module to be installed before another between two specific reboots, create two distinct installation stages between the reboots. For example, if your first software module copies files on the target and the second one runs a command on these files, you must place the first software module in an installation stage which occurs before the one in which you run the command software module.

1. To schedule the application of software modules, go to **Server** > **OS deployment** > **Software modules**. This opens a dialog window that allows you to order the different software modules stored on your OS deployment server. The dialog shows the different steps of a deployment with disk partitioning (in green), OS installation (in purple) and reboots (in red). Software components can be installed in between all of these steps, where they are placed inside the expandable installation stages (in yellow).
2. You can add, move, and delete reboot sequences by using the buttons at the bottom of the dialog window. You can also rename software installation stages.
3. You can expand the software installation stages to view their content by clicking on the **+** icon. You can then move individual software modules from one stage to another by drag-and-drop. The destination stage does not need to be expanded.

   Note: Drag-and-drop is limited to the **Software Application Order** window. You cannot drag-and-drop an item from the Software Module page.

Starting from 7.1.1 Fix Pack 6 the capability to run a software module has been added before **Disk partitioning** and before **OS installation**.

Before partitioning the disk you can:

- Copy the `/home` directory remotely.
- Capture the previous disk.
- Update firmware (not in scripting toolkit).
- Enable swap.
- In general, perform any other action that you want to run under the Linux deployment engine environment before the hard disk is altered.

Note: The following limitations apply to Linux software modules run before partitioning the disk:

- / folder is the ramdisk path, software with relative paths are copied on the ramdisk.
- The space available depends on the memory of the target (ramdisk limitation).

Before installing the operating system you can:

- Use it mainly for workaround and debugging purposes.

Note: The following limitations apply to Linux software modules run before installing the operating system:

- Working drive is always the / folder (ramdisk path), software with relative paths are copied on the hard disk. It is not possible to copy them directly on \ of Linux deployment engine because of space limitation and also because this is done in a different deployment phase than the one in which the command is running. Files on \ of Linux deployment engine would be lost.

Note: The new stages do not work for the VMWare and HTTP unattended setup. In this case, it is recommended to use the pre and post scripts. For the setup deployment, the after partitioning pass is displayed after a reboot.

When creating a recovery CD or exporting a RAD file, the software application order is automatically included.

Working with hardware configurations

It is sometimes necessary to run configuration tasks on the targets before installing the operating system, for example to update the firmware or to configure RAID volumes.

To automate this kind of operation with the product, you must perform a *hardware configuration task*, which uses a *hardware configuration object* stored on the OS deployment server. To create a hardware configuration object, you must have already created a *hardware environment*. This hardware environment contains WinPE or DOS files, updated with drivers specific to given hardware models and vendor-specific tools to perform hardware configuration tasks.

The hardware configuration tasks that you can perform with the product are

- RAID configuration
- Firmware update (BIOS and UEFI)
- Firmware settings (BIOS and UEFI)
- Hardware custom configuration, that is, any kind of tool that you can load into the environment and run from a command line.

You can also perform an inventory of RAID or Fiber Channel hardware.

Hardware configuration tasks are available only for targets with an x86 or an x86-64 architecture.

Example

To configure hardware with the product, for example a BIOS update with WinPE2 on an IBM target, you need to follow a number of steps.

1. Create a hardware environment with drivers and tools:
   a. Download Windows Automated Installation Kit (WAIK) from Microsoft and install it to have the WinPE2 files available.

   b. Download the latest ServerGuide scripting toolkit from IBM and extract it, for example, in directory `C:\IBM-SGTSK-WinPE2.x`.
   c. Run the **`SGTKWinPE.cmd`** command to prepare the WinPE2 environment with the needed IBM drivers. It creates the `.\sgdeploy\WinPE_ScenariosOutput\Local\RAID_Config_Only\ISO` directory, which contains both the WinPE2 binaries and the vendor-specific tools.
   d. Create a hardware environment with the hardware environment wizard.
2. Create a hardware configuration object with the hardware configuration wizard:
   a. Select **BIOS update** as the type of hardware configuration to be performed.
   b. Associate the hardware environment of step 1 on page 146 and your hardware model to the new hardware configuration object you are creating.
   c. Indicate the location of the BIOS update material, that is, a set of files containing in particular `wflash.exe`.
3. Perform the actual configuration task by deploying the hardware configuration object of step 2 on your target:
   a. Select a target (or several) in the Target Monitor.
   b. Select **Deploy now** in the contextual menu.
   c. Select **Perform hardware configuration tasks** and optionally other deployment tasks in the deployment wizard.
   d. Select the hardware configuration object that you want to apply and follow the remaining instructions of the wizard.

The hardware environment now runs as a ramdisk on the target, and, using vendor-specific tools, the BIOS is updated.

Setting up your environment

To perform hardware configuration tasks, you must set up a hardware-specific environment containing the vendor-specific scripting toolkit tools and the necessary drivers to run correctly (for example, network connectivity) on the target.

The hardware environment supported are those running scripts and tools in:

- WinPE 3.x
- WinPE 2.x
- WinPE 1.x
- DOS

Every environment is very specific to its vendor, and must be prepared with the suitable drivers and scripting toolkit tools.

WinPE3, WinPE2, WinPE1, and DOS cannot perform hardware configuration tasks (for example, RAID configuration or BIOS setting) by themselves. They must contain drivers to access the hardware and tools to perform the configurations. These drivers and tools are vendor-specific and vary for each type of target model. When you create an environment with the OS deployment server, you associate either WinPE3, WinPE2, WinPE1, or DOS, to vendor-specific drivers and tools. You can then associate the resulting environment to a specific set of target models and a type of hardware configuration tasks to create a hardware configuration object.

Because a hardware environment is run as a ramdisk, it does not leave any trace on the target after the hardware configuration task is performed.

64-bit WinPE 3.1 is required for performing hardware configurations tasks in UEFI mode.

Hardware configuration objects and tasks

A hardware configuration object is the association, on an OS deployment server, of a vendor-dependent environment, target models, a type of hardware configuration to be performed, and possibly some other commands. A hardware configuration task is performed at deployment time by loading and running the associated hardware configuration object containing a vendor-dependent environment on the target, before installing the operating system.

Hardware configurations tasks do not impact the following operating system deployment because Tivoli Provisioning Manager for Images configures the hardware through actions run in a ramdisk before the deployment of the operating system.

The execution flow is similar, regardless of the environment to run, or the type of hardware environment task:

1. The environment is loaded in memory, as a ramdisk
2. Any additional binary or configuration files are added to the ramdisk, based on the selection made in the web interface when creating the hardware configuration object
3. The computer boots the ramdisk
4. The hardware configuration task is run
5. The computer reboots
6. Tivoli Provisioning Manager for Images resumes the deployment sequence if any was selected, but a hardware configuration object can be run also as an independent task

The following types of hardware configuration objects are available:

RAID configuration
The hardware configuration wizard allows you to create a hardware configuration object to configure RAID adapters in a vendor-independent way. Tivoli Provisioning Manager for Images builds the vendor-specific configuration file.

BIOS update
The hardware configuration wizard allows you to create a hardware configuration object to update the BIOS firmware on the target.

BIOS settings
The hardware configuration wizard allows you to create a hardware configuration object to update the BIOS or BMC (baseboard management controller) settings through an initialization file.

Hardware custom configuration
The hardware configuration wizard allows you to create a hardware configuration object to perform any kind of hardware configuration. Any tool used for preparing the environment can be packaged in a custom hardware configuration object, injected into the ramdisk and run using command lines.

Capture hardware parameters
This option is available only if you do not already have a hardware capture configuration object.

The hardware configuration wizard allows you to create a hardware configuration object to capture RAID and Fiber Channel information from a target.

RAID and Fiber Channel hardware capture

Capturing RAID and Fiber Channel information requires the use of a vendor-specific environment.

Target inventory for CPU, memory, logical disks, PCI devices, motherboard, and so on, is managed by the OS deployment engine and all information is available immediately if requested. To complete the hardware target inventory with RAID and Fibre Channel information you need the vendor-specific scripting toolkit tools. The hardware capture is done in a similar way to that of the hardware configurations, which means that you need to load the vendor-dependent environment on the target to start the specific capture tool.

The captured hardware information for Fibre Channel and RAID disks can then be seen from the web interface:

Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Creating a hardware environment

To perform hardware configuration tasks on targets, you must have a hardware environment on your OS deployment server.

Before you can create your environment, you must prepare the files on the OS deployment server.

Instructions are provided for preparing the files using scripting toolkits for IBM, Dell, or HP products. It is recommended that you download the latest WinPE 3.x compatible scripting tool environments and use this version. However, the instructions for WinPE 2.x, WinPE 1.x and DOS are also provided.

IBM **IBM ServerGuide Scripting Toolkit WinPE 3.x based**

The following procedure is valid for BIOS targets:

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_utl_sgtkwin_2.30_windows_32-64.zip`.
2. Extract the toolkit into a local directory, for example, `c:\IBM-SGSTK-WinPE3.x`
3. As described in the User's Guide available in `c:\IBM-SGSTK-WinPE3.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, perform the following steps:
   a. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
   b. Install Windows AIK.
   c. Restart your computer.

   d. Expand files `ibm_utl_tsep_2.00_winpe_i386.zip` and `ibm_utl_tsep_2.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE3.x`
   e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
   f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x86.ini` as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

      ```
      SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x86.ini
      ```

      A directory `.\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO` is created and contains the environment tools.

The following procedure is valid for UEFI targets:

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_utl_sgtkwin_2.30_windows_32-64.zip`.
2. Extract the toolkit into a local directory, for example, into `c:\IBM-SGSTK-WinPE3.x`
3. As described in the User's Guide in `c:\IBM-SGSTK-WinPE3.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, you must then do the following:
   a. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en. and http://www.microsoft.com/download/en/details.aspx?id=5188.
   b. Install Windows AIK and supplements by following the Microsoft instructions provided on the download page.
   c. Restart your computer.
   d. Expand files `ibm_utl_tsep_2.00_winpe_i386.zip` and `ibm_utl_tsep_2.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE3.x`
   e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
   f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x64.ini` as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

      ```
      SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x64.ini
      ```

      A directory `.\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO` is created and contains the environment tools.

IBM IBM ServerGuide Scripting Toolkit WinPE 2.x based

1. Download the latest ServerGuide scripting toolkit from the IBM website. The name of the downloaded file is similar to `ibm_sw_sgtkw_2_1_windows_i386.zip`.
2. Extract the toolkit into a local directory, for example, into `c:\IBM-SGSTK-WinPE2.x`
3. As described in the User's Guide in `c:\IBM-SGSTK-WinPE2.x/sgdeploy/SGTKWinPE/Docs/UserGuide.pdf`, you must then do the following:
   a. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008. Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
   b. Install Windows AIK.
   c. Restart your computer.
   d. Expand files `ibm_utl_sep_1.00_winpe_i386.zip` and `ibm_utl_sep_1.00_winpe_x86-64.zip` located in `.\sgdeploy\updates\uxsp` into the directory in which the toolkit was extracted, for example `c:\IBM-SGSTK-WinPE2.x`
   e. Run `InstallSEPs.cmd` to install the System Enablement Pack.
   f. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers. Use the option `/Image` to exclude ISO and provide `ScenarioINIs\Local\Raid_Config_Only_x86.ini` if you use a 32-bit WinPE2, or `ScenarioINIs\Local\Raid_Config_Only_x64.ini` if you use a 64-bit WinPE2, as properties file to include all RAID and Fibre tools and to exclude all network tools. The command finds where the Windows AIK is located by itself.

      ```
      SGTKWinPE.cmd /Image ScenarioINIs\Local\Raid_Config_Only_x86.ini
      ```

      A directory `.\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO` is created and contains the environment tools.

IBM

IBM ServerGuide Scripting Toolkit WinPE 1.x based

1. Download the latest ServerGuide scripting toolkit from the IBM website.
2. Extract the toolkit into a local directory, for example, `c:\IBM-SGSTK-WinPE1.x`.
3. As described in the User's Guide in `c:\IBM-SGSTK-WinPE1.x\sgdeploy\SGTKWinPE\Docs\UserGuide.pdf` you must then complete the following steps:
   a. Download WinPE 2005.
   b. Run `SGTKWinPE.cmd` to create a WinPE image with the requested drivers for IBM servers.

IBM

IBM ServerGuide Scripting Toolkit DOS based

1. Download the latest ServerGuide scripting toolkit from the IBM website
2. Extract the toolkit into a local directory, for example, `c:\IBM-SGSTK-DOS`.

Note: DOS tools are deprecated. They are used only to support some older hardware.

Dell

Dell DTK Scripting Toolkit WinPE 3.x based

The following procedure is valid for BIOS targets:

To set up the WinPE 3.x environment for your Dell servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in English. Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to `DTK3.2.1-WINPE-22.exe`.
5. Extract the download file. For example, extract the file to the location `c:\ Dell-DTK-3.2.1`.
6. As described in the Dell User's Guide, in `C:\Dell-DTK-3.2.1\Dell\Docs\DTKUG.pdf`, you must then complete the following tasks:
   a. Open a command prompt in the directory containing the driver installation batch for `WinPE3.x: WINPE3.0_driverinst.bat`. For example, the directory, `C:\ Dell-DTK-3.2.1\Dell\x32\Drivers\winpe3.x`.
   b. Launch the file called `WINPE3.0_driverinst.bat <WINPEPATH> <DTKPATH>`, where *<WINPEPATH>* is the destination path to create the directory structure for WinPE 3.0 and *<DTKPATH>* is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called `WINPE3.0_driverinst.bat C:\Dell-DTK-3.2.1\WinPE3.x_Out_x32 C:\Dell-DTK-3.2.1\Dell\x32\drivers`. Launching this file preinstalls the Dell drivers into `boot.wim`.

The following procedure is valid for UEFI targets:

To set up the WinPE 3.x environment for your Dell servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in English. Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en. and http://www.microsoft.com/download/en/details.aspx?id=5188.
2. Install Windows AIK and supplements by following the Microsoft instructions provided on the download page.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to `DTK3.2.1-WINPE-22.exe`.
5. Extract the download file. For example, extract the file to the location `c:\ Dell-DTK-3.2.1`.
6. As described in the Dell User's Guide, in `C:\Dell-DTK-3.2.1\Dell\Docs\DTKUG.pdf`, you must then complete the following tasks:
   a. Open a command prompt in the directory containing the driver installation batch for `x64 WinPE3.x: WINPE3.0_driverinst.bat`. For example, the directory, `C:\ Dell-DTK-3.2.1\Dell\x64\Drivers\winpe3.x`.

   b. Launch the file called `WINPE3.0_driverinst.bat` *`<WINPEPATH>`* *`<DTKPATH>`*, where *<WINPEPATH>* is the destination path to create the directory structure for WinPE 3.0 and *<DTKPATH>* is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called `WINPE3.0_driverinst.bat C:\Dell-DTK-3.2.1\WinPE3.x_Out_x64 C:\Dell-DTK-3.2.1\Dell\x32\drivers`. Launching this file preinstalls the Dell drivers into `boot.wim`.

Dell

Dell DTK Scripting Toolkit WinPE 2.x based

To set up the WinPE2 environment for your Dell servers:

1. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008.Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest DTK scripting toolkit from the Dell website. The name of the downloaded file is similar to `DTK2.6-WINPE-56.exe`.
5. Extract the download file. For example, extract the file to the location `c:\ Dell-DTK-2.6 5`.
6. As described in the Dell User's Guide, in C:\Dell-DTK-2.6\Dell\Toolkit\Docs\DTK25UG.pdf, you must then complete the following tasks:
   a. Open a command prompt in the directory containing the driver installation batch for `WinPE2.x: VPE_driverinst.bat`. For example, the directory, `C:\ Dell-DTK-2.6\Dell\Drivers\winpe2.x`.
   b. Launch the file called `VPE_driverinst.bat` *`<WINPEPATH>`* *`<DTKPATH>`*, where *<WINPEPATH>* is the destination path to create the directory structure for Windows PE 2.0 and *<DTKPATH>* is the path to the Dell drivers in the extracted DTK toolkit. For example, the file might be called `VPE_driverinst.bat C:\Dell-DTK-2.6\WinPE2.x_Out C:\Dell-DTK-2.6\Dell\drivers`). Launching this file preinstalls the Dell drivers into `winpe.wim`.
7. Copy and rename the customized `C:\Dell-DTK-2.6\WinPE2.x_out\winpe.wim` to `C:\Dell-DTK-2.6\WinPE2.x_Out\ISO\sources\boot.wim`.

Dell

DELL Scripting Toolkit WinPE 1.x based

Note: Windows PE 2005 must be built from a Windows 2003 server for the Dell tools to work.

To set up the WinPE1 environment for your Dell servers:

1. Obtain a Windows PE 2005 file structure.
2. Copy it into a temporary folder, for example, `c:\winpe-dell`
3. The Windows PE 2005 directory structure should contain a directory named `I386` or `MININT`. If it contains a directory named `MININT`, rename it to `I386`.
4. Download the Deployment Toolkit from Dell.
5. Run the executable package to extract the toolkit to the disk of the OS deployment server. In the examples, it is assumed that you have extracted the toolkit into `c:\DELL-DTK`, which implies that you have a folder named `C:\DELL-DTK\Dell\Toolkit`.

6. To install the appropriate drivers for Dell servers in your WinPE image, follow the instructions of the DTK User Guide (*Running Deployment Scripts Using DTK and Windows PE*).

   You must:

   a. Install the drivers with the `driverinst.bat` script

   b. Modify `winpeoem.sif` and `winbom.ini`

   c. Add the RPC DLLs to the Windows PE directory.

      Note: Add the RPC DLLs in `i386\system32` instead of those in the Tools folder.

7. To verify that the drivers are installed, check for the file called `c:\temp\winpedell\i386\system32\racsvc.exe`.

HP

HP SmartStart Scripting Toolkit WinPE 3.x based

To set up the WinPE 3.0environment for your HP servers:

1. Download Windows Automated Installation Kit (AIK) for Windows 7 in EnglishWindows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest SmartStart Scripting Toolkit from the HP website: http://h18013.www1.hp.com/products/servers/management/toolkit/. The name of the downloaded file is similar to `SP47335.EXE`.
5. Extract the file into a directory, for example, `C:\HP-TK`.
6. As described in the *HP SmartStart Scripting Toolkit Windows Edition User Guide.pdf* in `C:\HP-TK\SWSetup\SP47335\` and the *Windows Preinstallation Environment User's Guide* (WinPE.chm) contained in Windows AIK, you must then mount the WinPE3.x base image for specific customization. For example, activate extra packages, add drivers, and so on.

   a. From the Windows AIK tools folder, run the command to create WinPE customization directory.

   ```
   C:\Program Files\Windows AIK\Tools\PETools>copype.cmd x86
   C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP
   ```

   b. Mount the base image launching **`Dism`** from the WinPE3.x_HP folder.

   ```
   Dism /Mount-Wim /WimFile:.\winpe.wim /index:1 /MountDir:.\mount
   ```

   c. Install the *neutral* WMI packages in the image.

   ```
   Dism /image:.\mount /Add-Package
   /PackagePath:"C:\Program Files\Windows AIK\Tools\PETools\x86
   \WinPE_FPs\winpe-wmi.cab"
   ```

   Enter the command on one line, although it does not fit on this example.

   d. Install also the language specific WMI package in the image.

   ```
   Dism /image:.\mount /Add-Package
   /PackagePath:"C:\Program Files\Windows AIK\Tools\PETools\x86
   \WinPE_FPs\en-us\winpe-wmi_en-us.cab"
   ```

Enter the command on one line, although it does not fit on this example.

e. Add the required drivers (.inf files) to the base image by using the `/Add-Driver` option of the `Dism` command.

```
Dism /image:<mounted image> /Add-Driver /Driver:<driverpath>
/Recurse
```

Where <*driverpath*> is the location of the .inf files found in the extracted drivers within the hpDrivers folder and `/Recurse` is an option to query all the drivers in subfolders.

```
Dism /image:.\mount /Add-Driver
/Driver:C:\HP-TK\SWSetup\SP47335\hpDrivers\Winpe30 /Recurse
```

Enter the command on one line, although it does not fit on this example.

f. Copy the `hpsstkio.sys` Toolkit I/O driver (required for the `conrep` and `rbsureset` utilities) from the HP driver directory to the Windows driver directory. For example:

```
copy C:\HP-TK\SWSetup\SP47335\hpDrivers\Winpe30\system\hpsstkio
\hpsstkio.sys C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\mount\Windows
\System32\drivers
```

Enter the command on one line, although it does not fit on this example.

g. Unmount the customized image to build the customized WinPE.wim:

```
Dism /Unmount-Wim /MountDir:.\mount /Commit
```

7. Copy and rename the customized file

```
C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\WinPE.wim
```

into

```
C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\ISO\sources\boot.wim
```

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

HP

HP SmartStart Scripting Toolkit WinPE 2.x based

To set up the WinPE2 environment for your HP servers:

1. Download the Windows Automated Installation Kit (AIK) 1.1 32-bit in English. For Windows Vista SP1 and Windows Server 2008.Windows Automated Installation Kit (AIK) 1.1 is distributed by Microsoft and is available on the Microsoft website from the following link: Windows Automated Installation Kit (AIK).
2. Install Windows AIK.
3. Restart your computer.
4. Download the latest SmartStart Scripting Toolkit from the HP website: http://h18013.www1.hp.com/products/servers/management/toolkit/. The name of the downloaded file is similar to `SP38836.EXE`.
5. Extract the file into a directory, for example, `C:\HP-TK`.
6. As described in the *HP SmartStart Scripting Toolkit Windows Edition User Guide.pdf* in `C:\HP-TK\SWSetup\SP38836\` and the *Windows Preinstallation Environment User's Guide* (WinPE.chm) contained in Windows AIK, you

must then mount the WinPE2.x base image for specific customization. For example, activate extra packages, add drivers, and so on.

   a. From the Windows AIK tools folder, run the command to create Windows PE customization directory. For example: `C:\Program Files\Windows AIK\Tools\PETools>copype.cmd x86 C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP`)
   b. Mount the base image launching imagex from the WinPE2.x_HP folder. For example, `imagex /mountrw WinPE.wim 1 .\mount`.
   c. Install the WMI packages in the image: `peimg /image=.\mount /install=*WMI*`
   d. Add the required drivers (.inf files) to the base image by using the `peimg /inf` command.

      ```
      peimg /inf=<driverpath> .\mount
      ```

      Where *<driverpath>* is the location of the .inf files found in the extracted drivers within the hpDrivers folder. For example, `peimg /inf=c:\HP-TK\SWSetup\SP38836\hpDrivers\Extr-Drivers\nic\b06nd .\mount`.
   e. Repeat step d. for each additional device driver.
   f. Copy the `hpsstkio.sys` Toolkit I/O driver (required for the `conrep` and `rbsureset` utilities) from the HP driver directory to the Windows driver directory. For example:

      ```
      copy C:\HP-TK\SWSetup\SP38836\hpDrivers\system\hpsstkio\hpsstkio.sys
      C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\mount\Windows\System32\drivers
      ```
   g. When you finish customizing the image, prepare the environment image by using the `peimg /prep` command:

      ```
      peimg /image=.\mount /prep
      ```
   h. Unmount the customized image to build the customized WinPE.wim:

      ```
       imagex /unmount /commit .\mount
      ```
7. Copy and rename the customized `C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\WinPE.wim` file into `C:\HP-TK\SWSetup\SP38836\WinPE2.x_HP\ISO\sources\boot.wim`.

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

HP

HP SmartStart Scripting Toolkit WinPE 1.x based

The initial setup for the HP SmartStart Scripting Toolkit is similar to the setup of the Dell Hardware Toolkit, because both Toolkits require Windows PE. Some details are not included, but you can read them in the Dell section.

1. Download the Win32 HP SmartStart Scripting Toolkit version of the toolkit on the HP website.
2. Extract it to the disk of the OS deployment server, for example, in `c:\HP-TK`.
3. Create a Windows PE 2005 folder for the HP tools:
   a. Copy a Windows PE file structure to a temporary folder (`c:\winpe_hp`)
   b. Install the HP drivers in the Windows PE directory, as explained in the User Guide for the HP Hardware Toolkit
      1) Run the executable file under hpDrivers

2) Give the location of the i386 folder of your Windows PE folder

Note: In RAID configuration tasks, the size specified in percentage (%) is not supported.

To create your environment, perform the following steps:

1. Go to **Server** > **Advanced features** > **Hardware configurations**.
2. Click **New environment** and follow the wizard instructions. You must
   a. Ensure that the web interface extension is running on the computer where Windows AIK and the environment tools have been prepared.
   b. Provide the path of the folder in which the environment tools are located, that is where you have installed the scripting toolkit. For example:

      IBM

      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO (BIOS)`
      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO (UEFI)`

      Dell

      - `C:\Dell-DTK-3.2.1\Dell\x32 (BIOS)`
      - `C:\Dell-DTK-3.2.1\Dell\x64 (UEFI)`

      HP `C:\HP-TK\SWSetup\SP47335`
   c. Provide the path of the folder in which the environment material is located, that is the WinPE files. For example:

      IBM

      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x86\ISO (BIOS)`
      - `C:\IBM-SGSTK-WinPE3.x\sgdeploy\WinPE_ScenarioOutput\Local_Raid_Config_Only_x64\ISO (UEFI)`

      Dell

      - `C:\Dell-DTK-3.2.1\WinPE3.x_Out_x32\ISO (BIOS)`
      - `C:\Dell-DTK-3.2.1\WinPE3.x_Out_x64\ISO (UEFI)`

      HP `C:\HP-TK\SWSetup\SP47335\WinPE3.x_HP\ISO`

To view the created environment go to **Server** > **Advanced features** > **Hardware configurations**. Alternatively, you can also go to **Server** > **OS deployment** > **Software modules**. To view the created environment, look under a specific environment folder.

Now, you can create hardware configurations using this environment.

Creating a hardware configuration object

A wizard allows you to easily create hardware configuration objects.

Before you can create a hardware configuration object, you must have created the environments needed to later perform the hardware configuration tasks.

1. Go to **Server** > **Advanced features** > **Hardware configurations**.
2. Click **New hardware config.**.
3. Select the kind of hardware configuration that you want to create.

4. Provide at least one target model and environment pair on which the hardware configuration can apply.
5. For BIOS update, BIOS settings, or Hardware custom configuration the specific files or set of files can be downloaded from the specific vendor sites.
6. Follow the wizard instructions.

To view or edit a hardware configuration, select the hardware configuration and select **View configuration details** in the contextual menu. In the **Hardware configuration details**, use the **Edit** buttons to update the different sections.

Creating a hardware capture configuration

A wizard allows you to easily create hardware capture configuration in a way similar to that for hardware configurations.

Before you can create a hardware capture configuration, you must have created the environments needed to later run the hardware capture.

- If you do not yet have a hardware capture configuration, perform the following steps:
  1. Go to **Server** > **Advanced features** > **Hardware configurations**.
  2. Click **New hardware config.**
  3. Select **Hardware discovery**.
  4. Provide at least one target model and environment pair on which the hardware capture can apply.
  5. Follow the instructions of the wizard.
- If you already have a hardware capture configuration, you can add target model and environment pairs, as follows:
  1. go to **Server** > **Advanced features** > **Hardware configurations**.
  2. Select **Hardware discovery**.
  3. Double-click **Hardware capture configuration**.
  4. Under **Hardware environment matching**, click **Edit**.
  5. Click **Add a new line** and select the model and environment values
  6. Repeat step 5 for each pair to be added.
  7. Click **OK**.
  8. Click **Back** to return to **Server** > **OS deployment** > **Hardware configurations**.

To view or edit the hardware capture configuration, go to **Server** > **Advanced features** > **Hardware configurations**. Select **Hardware discovery**, and double-click the hardware capture configuration. In the **Hardware configuration details** page, click **Edit** to update the different sections.

You can now capture RAID or Fiber Channel information.

Capturing hardware information using templates

When you capture hardware information with templates, this capture is done every time the template is used.

Capturing hardware information with templates requires an additional reboot to boot the specific hardware configuration environment (WinPE, DOS,...) and launch the specific scripting toolkit tools.

Note: You can not capture hardware information from a target started with a network boot media.

Capturing hardware information with templates always tries to capture both RAID and Fiber Channel. To run the capture:

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select **Idle Layout** or **Deployment Schemes**, depending on which state you want to perform the hardware capture. If you select **Deployment Schemes**, the discovery is performed at deployment time.
3. Double-click the chosen template to view its details.
4. Click **Edit** on **General settings**.
5. Under **Perform inventory on:**, select **RAID**. Select this option in the deployment scheme only if you are creating a hardware configuration for the hardware capture. In this way you avoid a failure at any target PXE boot.
6. Click **OK**.

Capturing hardware information once

When you want to capture hardware information only once for a target, or a group of targets, you do this with a specific tool.

Capturing hardware information requires an additional reboot to boot the specific hardware configuration environment (WinPE, DOS,...) and launch the specific scripting toolkit tools.

Note: You can not capture hardware information from a target started with a network boot media.

1. Go to **Server** > **OS deployment** > **Target Monitor**.
2. Select a target or a group of targets.
3. Select **Additional features** from the contextual menu.
4. Double-click the chosen template to view its details.
5. Select **Capture hardware parameters** and click **Next**.
6. Select **Raid capture**, **Fiber channel capture**, or both, and click **Next**.
7. Follow the instructions of the wizard.

When captured, the RAID and Fiber channel information can be viewed. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. On this page look under the **Inventory** tab.

Task templates for Linux operating systems

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout
The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout
The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout
The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout
The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Customizing a screen layout

You can customize the screen layout of a target.

To customize a screen layout:

1. Select the layout that you want to customize in the right pane of the **Task Templates** page of the web interface page.

   Note: An actual layout must be selected and not a layout folder (left pane)
2. At the bottom of the page, the screen layout is shown in reduced size. Click **Customize GUI** to open the screen layout editor.
3. The editor is composed of a left column, containing instructions, a *What-You-See-Is-What-You-Get* (WYSIWYG) view of the screen being edited and a bottom banner with action buttons.
4. Click on the action buttons or directly on the items that you want to modify to see their editable properties displayed in the left column. Make the wanted changes and then click **Save** to keep your new screen design. Return to the **Task Templates** page by clicking **Back**.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard. Go to **Server** > **OS deployment** > **Task templates**. and click **New deployment scheme**.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section. Some advanced deployment scheme features are available only in this mode and not through the wizard.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   The following parameters apply for simple one-time deployments and for redeployment operations.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, so the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters

   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

Edit parameters for targets that are not yet in the database
to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

Never edit parameters
to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

Yes, display a warning
if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment
if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, because Tivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of targets to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed target, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, / on UNIX operating systems or `c:/` on Windows operating systems. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: Multicast is available only if:

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of targets that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast

to deploy targets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast

- **without explicit synchronization**

to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.

- **with synchronization on all targets**: to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.
- **with the following synchronization parameters:**

  to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

In addition to that, you can also configure two additional parameters:

- **`Multicast block size (MB)`** with default value 16
- **`Multicast TOC block size (KB)`** with default value 32

During a deployment configured to use multicast, the server splits the system profiles in blocks that are sent to all the targets in the same multicast group. Before sending each block, the server sends a packet, called ToC, that describes the content of a block. The parameter **`Multicast block size`** is the maximum size of a block of data. The second parameter **`Multicast TOC block size`** is the maximum size of the ToC.

These parameters can be configured to improve the performance of the multicast deployment. If you have a high speed network rated at 1 GB and a profile of 15 or 20 GB, it is recommended to change both the values to at least 64.

For Windows Vista and later, and Linux platforms, you can decide to use a network share on Windows, or Linux HTTP, to download the files to the targets , rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP provides a shorter installation time. For Linux deployments, you can also allow the use of the Linux Deployment Engine for 64-bit cloned profiles using PXE.

To use a network share on Windows or Linux HTTP:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter in the **Network share module** section. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module).

On Linux only, to use the Linux Deployment Engine:

- Select **Use Linux Deployment Engine x86-64 if applicable** to deploy a Linux cloned 64-bit profile using PXE, for driver injection purposes. This option is ignored in all other cases.

On-site deployment

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment option

Indicate if you want to keep the deployment image in a protected partition and the size of this partition. These options are valid only to configure the deployment scheme for redeployment. More information is available in deploy/tosd_redeplscheme.dita.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation

This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

This parameter, located in the **General settings** section, is set to **No** by default. Setting this parameter to **Yes** unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user. Review the section on network boot scenario of the deployment process topic.

Unbind software module at the end

This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is relevant for Windows Vista/2008/7/2012 deployments only.

Send mail at end
This option can be set only if a *sendmail* tunnel has been created.

Send mail to
This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network
This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button
This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Creating media for deployment for Linux operating systems

You can create deployment media such as CDs, DVDs, or USB drives to install machines without connecting them to the OS deployment server.

You can use this kind of deployment when there is no connection or connection to the OS deployment server is very slow.

Some typical situations are small branch offices with slow links and no local deployment server, isolated computers with no connection to an internal network, laptop users currently away from LAN or connected using a modem.

If the data you want to use does not fit on a single CD or DVD, use a USB drive.

You can also create offline Win PE-based deployment media in kernel-free mode. You select the deployment engine type and the architecture to use, as well as System Profiles and any software module you want to include in the deployment media. If you have more than one WinPE deployment engine, you can select the target models and the matching engines for these models are included. The engines are cached locally on the target.

Note:

- You must create the deployment media from an OS deployment server or a web interface extension installed on a computer with the same byte order (little endian or big endian) as the one on which you want to use the deployment media.
- To deploy Windows system profiles on Hyper-V, make sure that the boot order indicates the hard drive before the CD-ROM or USB drive.
- Redeployment is not available when deploying from a deployment media.

Deploying Linux from a deployment media

- When you use a deployment media to deploy a Linux operating system, the target keyboard layout cannot be changed.
- When you use a deployment media containing both Windows and Linux system profiles and you want to deploy a Linux system profile, you are asked twice to select your system profile.

Creating an OS deployment USB drive with the wizard

Tivoli Provisioning Manager for Images can automatically generate deployment USB drives that replay the deployment process for a given system profile or for any kind of software modules available.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must be formatted as FAT32 or NTFS.

Note: SuSE Linux Enterprise Desktop cloning is not supported on USB drive deployments.

Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The deployment USB drive is self-contained and can be used instead of a CD or DVD to provision a target entirely offline, without using the OS deployment server. These deployment USB drives can also be used to deploy computers without a PXE-compliant network adapter.

To create OS deployment USB drives:

1. Perform one of the following operations:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.
   - Go to **Server** > **OS deployment** > **Software modules**.
2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment USB key** to start the USB key wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.

6. (*Optional*) Change settings for targets running the USB key that you are creating.

   Included objects

   When selecting objects to be included, be aware that:

   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile must be included in your image.
   - One, and only one, deployment scheme must be included in your image. In this deployment scheme, do not select the **Download files with a network share when applicable** option in the **Network settings** section, because HTTP deployment is not available offline.
   - The software application order is automatically included.
7. If your USB key has already been used as a deployment media, you might choose to keep a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB key.
8. Plug your USB key into a machine running the web interface extensionand specify its address.
9. Choose the drive matching your USB key.
10. Click **Finish** to close the wizard.

Use the USB drive to deploy a given system profile or any kind of software module.

Creating an OS deployment USB drive with command lines

You can create an OS deployment USB drive that Tivoli Provisioning Manager for Images can use when a target cannot boot from the network.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must have boot capabilities and a FAT32 or NTFS filesystem. The drive must be already formatted; existing files on the partition are not deleted. USB keys already filled with a bootable operating system might not work.

Note: Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The command line must be used only when the web interface is either inappropriate or unavailable.

Use this command line:

- On Windows operating systems:

```
rbagent.exe -s <OSD_server_ip_address>:<OSD_server_password>
rad-usbget <drive>
keepshared|delshared preferwpe|prefermcp nodes
```

Where:

OSD_server_ip_address

Is the IP address of the OS deployment server.

`OSD_server_password`
Is the password for the administrative user (typically `admin`) on your OS deployment server.

`drive` Is a drive letter of the Windows target where you run the `rbagent` command. The rad-usbget command adds requested files to the FAT32 or NTFS partition and makes it bootable. The drive must be already formatted. Existing files on the partition are not deleted.

`keepshared`
Keeps a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB drive.

`delshared`
Deletes a shared repository of previous data.

`preferwpe|prefermcp`
Defines if the Linux deployment engine environment or WinPE is loaded from the USB drive, when a target boots from this USB drive, without accessing the network. Only when Linux deployment engine or WinPE is running, does the target connect to the network and try to contact an OS deployment server. If you deploy only Linux, specify `prefermcp` to skip WinPE. You can specify `preferwpe` only if there is a WinPE deployment engine on the OS deployment server.

nodes Defines the deployment settings with a space-separated list of objects. Specify at least `DEPLSET:Default` for the deployment schema, and `PROFILE:SystemID` for the system profile.

You can now boot the target using the OS deployment USB drive instead of the network card. To use the PXE emulation USB key, insert the USB key into the drive and restart the target. If your machine does not boot from the USB key, check the BIOS boot list to see if your optical drive is included in the boot sequence and is listed before the hard disk. Most machines also allow you to select the temporary boot device without changing the boot sequence in BIOS.

Creating OS deployment CD and DVD

Tivoli Provisioning Manager for Images can automatically generate deployment CDs and DVDs that replay the deployment process for a given system profile or for any kind of software modules available. You can use this feature to create OS deployment CDs and DVDs that can be easily sent through the Internet or by e-mail, to refresh a computer back to its initial working state after installation.

The CD/DVD deployment occurs in kernel mode. Microsoft tools are used to build the CD/DVD. By specifying the target models, the product automatically determines which deployment engine to use and the drivers corresponding to the specified target models are added to the CD/DVD. These CDs and DVDs can also be used to deploy computers without PXE compliant network adapter. The creation of DVDs and media spanning is supported. These media can be protected using an activation code preventing unauthorized personnel from using it.

To create OS deployment CD and DVD:

1. Perform one of the following operations:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.
   - Go to **Server** > **OS deployment** > **Software modules**.

2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment CD or DVD** to start the CD and DVD wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. Follow the wizard instructions to create an ISO image.

   Included objects

   When selecting objects to be included in the ISO image, be aware that:

   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile must be included in your image.
   - One, and only one, deployment scheme must be included in your image. In this deployment scheme, do not select the **Download files with a network share when applicable** option in the **Network settings** section, because HTTP deployment is not available offline.

     Note: You can deploy a Linux system profile using a network boot media only if the root partition is not LVM.
   - The software application order is automatically included.

   Hardware options

   In the hardware options settings some boot options can be customized. By default the options are unchecked but some special cases can require changes. In particular, if the CD or DVD is to be used on a USB drive or as a secondary drive, it might be necessary to specify the option **use BIOS for CD or DVD ROM access**. When this option is selected, on some hardware it might also be necessary to select **disable enhanced disk access** (for IDE CD or DVD) or **disable USB** (for USB CD or DVD) to ensure that Tivoli Provisioning Manager for Images use of other IDE or USB devices does not interfere with the BIOS access to the CD or DVD. In addition, deploying from the second CD or DVD drive of a target only works if you can ensure that subsequent boots keeps booting on the same CD or DVD drive.

   Security issues

   For security issues, you might want to protect deployment from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment to proceed.

   You might also want to hide the content of the ISO image that contains sensitive information such as product keys. To do this, select **Hide the content of CD or DVD** in the CD or DVD Wizard. If you then try to access files in your ISO image, you see the content as `CDROM_content_hidden`.

   Size of the ISO file

   The wizard allows you to choose the size of the ISO images.

   a. Enter the maximum size in the field displayed.

b. Click **Next** and the wizard starts to precompute the ISO file size.

The wizard displays the results for the number of disk images and the size required. You then have the option to:

- Download it directly from the server.
- Use the web interface extension
- Generate it on the server itself in the import directory.
- Generate it on another computer running the web interface extension

Note:

- When creating the ISO files, all objects of type *single file to copy*, *image headers*, and *WIM images* (which includes Windows Vista/2008/7/2012 unattended setup profiles), are put on the first CD or DVD. Therefore, the first ISO file might grow larger than the requested spanning size if the total size of the files to be put on the first ISO requires it.

  For example, if you try to create an OS deployment DVD containing both Windows Vista/2008/7/2012 unattended setup profiles, both profiles must be contained on the first ISO, but their total size is larger than 4 GB. Therefore, the ISO cannot be burned into a single layer DVD. In this case, either use a double layer DVD, or transfer the ISO without burning it.
- When deciding where to generate the ISO image, be aware that:
  - If the estimated size is bigger that 2 GB, do not use the link to download directly from the server, because of limitations of web browsers. An exception to this rule is Mozilla Firefox on Linux, which can extract files as large as 4 GB or more.
  - Because of file system limitations, do not extract files bigger than 4 GB on FAT32 partitions.

Use a CD creation tool to burn the ISO image onto disks.

Note: Vista 2008 Windows 7 Windows Vista/2008/7/2012 unattended setup profiles contain at least one file larger than 1 GB which cannot be split. Therefore, ISO files containing Windows Vista/2008/7/2012 unattended setup profiles must be burned on a DVD.

If you encounter problems when deploying from this CD or DVD on a virtual machine, make sure that the CD drive comes after the hard disk in the boot order.

Setting up an activation code

For security issues, you might want to protect deployment or booting from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment or the network boot to proceed.

To prevent being asked several times for the activation code during deployment:

- The deployment scheme included on your deployment CD must have the network setting **Use 'BIOS fall back MBR' to start PXE** set to **No**.
- The boot order of your target must be set to hard disk first and you must boot on the CD manually the first time.
- To set up an activation code for the first time, when creating the deployment CD:
  1. Select **Include activation code protection** in the deployment media wizard.

2. Enter and confirm the chosen password. You must remember this password if you want to obtain other activation codes for this CD.
3. Set a password expiration date under **Valid until**.

- To obtain a new activation code, for example, if you must use the CD after the current activation code expiration date:
  1. Click **Generate Media** on the Profiles page to start the deployment media wizard.
  2. Select **Generate a new activation code**.
  3. Click **Next** and follow the wizard instructions to obtain your new activation code. You must remember the password given when creating the first activation code for this CD.

The wizard provides you with the generated activation code that you need when using the CD.

Deploying Linux

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

The deployment process

In Tivoli Provisioning Manager for Images, a deployment is made up of several steps that are automatically run in sequence without user interaction:

1. Hardware configurations are optionally deployed, for example, to create RAID volumes.
2. Partitions are created on the hard disk, and then formatted according to information contained in the system profile.
3. All deployment objects (system profiles, partition files, and software modules) are downloaded to a temporary storage location on the hard disk.
4. Operating system files are written in the hard disk partitions, creating a bootable operating system with files and applications configured by database bindings between the *target* and *software modules*.
5. Target-specific configuration, such as the *host name* or the *product key* are gathered from the database to create a textual configuration file used by the system preparation tool.
6. The operating system is started, allowing LinPrep to configure the operating system according to information stored in the Tivoli Provisioning Manager for Images database.
7. Additional software is optionally installed, if it must be installed after the operating system.
8. The temporary storage location is cleaned. Installation files are removed.
9. Tivoli Provisioning Manager for Images takes control again when LinPrep has completed and rebooted the target, and displays a message indicating the status of the deployment.

When the deployment is complete, the operating system is installed and ready to be used by the user defined for this target in the database.

HTTP deployment

From version 7.1.1.3 of the product, the deployment of Linux system profiles in unicast can be performed through HTTP. By default, Linux setup deployment is performed using HTTP. Booting the Linux kernel, the kernel downloads the packages using HTTP. This results in a faster deployment and no third party is involved. In this way you also avoid that unallocated space is left on the target disk equal in size to the rembo cache used by the deployment. In non-HTTP deployment this space is not added to the partition defined as the 100% of the remaining disk space.

To activate or deactivate HTTP downloads in unicast, change the value of **Download files with a network share or Linux HTTP when applicable** in the **Network settings** section of your deployment scheme. By default, the value is set to **Yes**.

The deployment of Linux system profiles in multicast remains unchanged.

Network boot scenarios

Depending on the number of OS configurations bound to a specific target, a target behaves differently when it boots on the network:

- If no OS configuration is bound to the target (for example, when a target starts for the first time and has not been configured), a special screen is displayed that asks the administrator to configure an OS configuration binding for this target on the OS deployment server. Deployment is not possible until an OS configuration is bound to the target.
- If one or more OS configurations is bound to this target, but no deployment has been scheduled on the server, a screen is displayed with a list of all the OS configurations bound to the target. Clicking on an item in the list starts an interactive deployment for the selected OS configuration, using either the **Default** deployment scheme (if no deployment scheme has been configured for this target), or the deployment scheme used during the last deployment.
- If one or more OS configurations are bound to this target, and a deployment has been scheduled on the server for a specific OS configuration, the target immediately starts the deployment without requiring any user intervention.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Note: For Linux deployments on HP servers with two RAID arrays, edit the `ks.cfg` file and add the following entries:

```
clearpart --all --drives=cciss/c0d1
zerombr yes
```

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

When your target has different network card interfaces, before deploying a system profile, ensure you define the target network configuration by performing the following steps:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Click **Switch to Advanced IP settings mode** in the **Common networking info** section.
2. Click **Edit**.
3. Set the connection name in the **Connection name** field.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. If you do not choose this option, select the type or types of deployment you want to perform. You can install additional software only if you deploy an operating system.

a. If you have selected **Perform hardware configuration tasks**, indicate which hardware configurations you want to deploy.

5. Select **Simple deployment** and click **Next**
6. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Three options are available in the deployment wizard to deploy targets without physically interacting with the computers.

   - **Try to wake up targets currently powered off using WOL** asks the Target Monitor to send IBM Wake on LAN packets to wake up targets. Waking up targets only works on carefully designed modern computers. A target can only be woken up if its network adapter and its system board support Wake on LAN packets, and if the network adapter has been shut down properly. If the network adapter is not in the appropriate power state, Wake on LAN packets will not wake the computer up. This is not specific to Tivoli Provisioning Manager for Images, but is rather a general limitation of the Wake on LAN technology.
   - **Try to wake up targets using management interface** asks the Target Monitor to contact the targets and send a reboot request. If you are running the web interface extension that uses specific arguments starting with **rad-**, you might not be able to reboot targets remotely. They must be rebooted manually. You need the web interface extension running with the correct privileges to run a remote boot.
   - **Try to reboot targets running theweb interface extension** asks the Target Monitor to contact the targets if they are running under Windows and send a reboot request. If you are not running Windows, you cannot reboot targets remotely. They must be rebooted manually. If you are running Windows, you need the web interface extension running with the correct privileges to run a remote boot.

   If you have not selected one of these options or if they do not work, and if the target you are trying to deploy is not powered on, turn it on now and make it start on the network.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Deploying a hardware configuration

A wizard allows you to effortlessly deploy hardware configurations.

To start a hardware configuration deployment you must first have at least a hardware configuration environment and a hardware configuration.

Note: You can not deploy a hardware configuration from a target started with a network boot media.

1. Select a single target or multiple targets on the Target Monitor page. To do this go to **Server** > **OS deployment** > **Target Monitor**. To select multiple targets or deployment, select an **administrative group** , a **custom list**, a **subnet**, or click on **individual target** names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the second screen of the deployment wizard, you must select at least **Perform hardware configuration** tasks and if you want to proceed with the Operating System/Software deployments you can also select another option.
4. Select one or several hardware configuration(s) you want to apply on target. RAID Configurations, BIOS Settings, BIOS Updates or Hardware custom configurations are classified in a matching folders.
5. Follow the deployment Wizard as it is described in the chapter Deploying depending on the options you chose above.

Every configuration you selected will automatically use the appropriate environment and only be applied if the model is matching the target.

Redeploying

This function is a special deployment scheme that gives you the ability to rapidly restore an image to a computer from a hidden partition on the computers hard-disk.

During the original image deployment to the computer, Tivoli Provisioning Manager for Images creates a hidden partition on the hard-disk of the target computer. When it has finished deploying the master image on the computer, it stores a reference image into the hidden partition. It is possible to store one or more reference images into a hidden partition on the computer.

Note: Before running a deployment task on a machine with a redeployment partition, ensure you remove the hard disk partition content.

Each time the system is booted, either off the hard-disk or using network boot, Tivoli Provisioning Manager for Images intercepts the boot process of the computer and presents a customizable menu of possible actions. Those actions are:

- Boot the system off the current image on the hard-disk.
- Do a quick cleanup of the currently deployed image against the reference image and restore the image from the hidden partition.
- Do a format and full restore of the reference image from the hidden partition. Using this function, it is possible to effectively have a fresh image deployment every day for the optimum performance of a system.
- Choose and deploy another configuration available on the hidden partition. This option takes as long as the format and restore option.

Note: Redeployment is not available when deploying from deployment media or when booting in kernel-free mode.

The purpose of redeployment

A computer generally works the best and the fastest on the day that it is installed. At that time, the system is completely clean, free of any undesirable processor-consuming gadgets, and all programs are configured for their optimal use by the system administrator. The purpose of redeployment is to ensure that the system is reset to this optimal state at every boot (or at some fixed interval).

There are three categories of systems that experience the most visible need for the redeployment technology:

Public computers
such as schools, universities, and Internet cafes, where users cannot be relied on to preserve the computer integrity, because the computer is not their own

Critical systems
such as banks, insurance companies, and industrial plants, where the company cannot afford to risk computers being reconfigured or infected by malicious software

Embedded systems
such as ticket machines, airport information systems and ATMs, that must be quickly rebuilt to their original OS configuration, without using a specific infrastructure

Because redeployment often occurs at the user's desk, it is necessary to find a solution that is quick, easy to use, does not require any significant infrastructure, and does not affect the work process of other users. This rules out standard deployment tools, because they impose a significant load on the network and affect other users' ability to perform their tasks.

Note: The redeployment feature is not intended to be used on virtual machines. On virtual machines, you should leverage the snapshot feature of your hypervisor rather than use the redeployment feature.

The redeployment process

Redeployment involves several steps, including creating a reference image of the target, and saving it as a redeployment partition.

Redeployment steps

Tivoli Provisioning Manager for Images addresses the challenge of redeployment with the following steps:

- At the end of a deployment, Tivoli Provisioning Manager for Images creates a reference image of the target, and saves it into a protected redeployment partition (invisible to the user and to the operating system itself). This increases deployment time by roughly 10% compared to a simple deployment, as most of the files are already present as file archives on the disk at that time.
- Every time a target starts, Tivoli Provisioning Manager for Images hooks the boot process before the operating system starts (using PXE or a special Master Boot Record).
- If configured to do so, Tivoli Provisioning Manager for Images authenticates the user of the target against the server database to restrict the use or the maintenance of the target to authorized persons only.
- If configured to do so, Tivoli Provisioning Manager for Images offers the choice of several OS configurations available on the target (multiboot), and of several levels of "cleaning".
- Using the reference image saved during deployment, Tivoli Provisioning Manager for Images resynchronizes the hard-disk content to its reference state. This typically takes only a few seconds, but can take up to a few minutes if everything on the hard disk has been deleted.

Offline redeployment limitations

Offline redeployment behaves slightly differently from online redeployment as the OS deployment server cannot be contacted for information. These limitations are removed after the target contacts the OS deployment server again. For example, interrupted tasks are not automatically resumed and changes to the partition scheme cannot be recovered.

Moreover, authentication with offline redeployment does not work. A message warns the user.

Note: If you plan to use redeployment with multiple OS configurations offline, make sure that all the preloaded OS configurations have exactly the same partition layout (number and size), because Tivoli Provisioning Manager for Images cannot create new partitions offline or to resize existing partitions offline. Failure to do so prevents you from redeploying offline some of the preloaded OS configurations.

Redeployment with multiple operating systems

You can preload up to three operating systems on a target, with a menu allowing the user to select which operating system to start.

Scenario

You want to provision the computers of a classroom with two different operating systems (for example, Windows 2008 R2, by unattended setup, and RHEL by cloning). When entering the classroom, the student must choose between the two operating systems. For security reasons, you want to make sure that the operating system which is started is always in a clean state. You also want the selected operating system to install and start quickly.

Principles

To achieve this, you must install each operating system in its own partition, save the OS configurations in a protected partition. Before you start an operating system, you do a rapid verification of the operating system partition with the information in the protected partition.

Requirements

For you multiple operating systems to cohabit in a single target and to be able to start them individually, you must follow these guidelines strictly:

- The hard disk of the targets must be large enough to contain the three operating systems and the protected partition.
- You must create a separate system profile for each operating system.
- All the profiles must have the same number of partitions, in the same format.
- Each operating system must be in a distinct partition, and all other partitions must be empty during the system profile creation.
- Each operating system must be in a primary partition, and there is a maximum of three primary partitions.
- In the system profiles, partition numbers cannot be modified.
- An offline refresh does not update the partition table.

Before you start, select the target machine to be used to create the system profiles needed. For example, this procedure can be executed with a hard disk of 40 GB, and 12 GB are assigned to each operating system.

Procedure

1. 1. Create your Windows 2008 R2 unattended system profile.
   a. From the web interface, create a new unattended system profile with the profile wizard, following the instructions.
      1) Create one primary and bootable NTFS partition for Windows 2008 R2 and determine the right size for this partition.
   b. Customize your OS configuration.
      1) Set the administrator name in the configuration.
      2) Optionally, bind software modules.
   c. Deploy a target with this profile.
2. Create your RHEL cloning system profile.
   a. Start the target where you deployed the Windows 2008 R2 with the RHEL installation CD.
   b. In the Linux installer, in the partitioning steps, select the option "Create custom layout, and modify the layout as follows:
      1) Leave unchanged the first NTFS partition.
      2) Create one primary partition (EXT3) for / (partition 2) and determine the right size for this partition.
      3) Create another primary partition of 1 GB for Linux swap.
      4) Install the GRUB loader.
   c. Install RHEL in partition 2.
   d. Customize your installation.
   e. After the installation, from the web interface, clone your target to create a new linux system profile.
   f. After the clone, check the partition layout retrieved by the OSD. In particular check and correct if needed the following settings:
      1) For the NTFS partition:
         a) Edit the size to the right value.
         b) Set the bootable flag to no.
      2) For the Linux EXT3 partition:
         a) Set the bootable flag to yes.
      3) For the Linux swap partition:
         a) Set the size to the right value (1 GB)
3. Update the two OS configurations.
   a. Edit the partition scheme for each OS configuration so that partitions have the same size on each OS configuration.
      1) For the unattended Windows 2008 R2 system profile you must create two Linux partitions, one EXT3 primary not bootable for / and one Linux swap primary of 1 GB.
   b. Use the best size found for each operating system.
   c. Set the options Must be deployed and Must be redeployed so that only the partition containing the operating system (and also the swap partition in the Linux case) it is actually deployed or redeployed for each system profile.
4. Test each system profile. Each operating system installs in the correct partition, without impacting other partitions.
5. Create a specific deployment scheme for this redeployment.

a. Export the two system profiles into a RAD file. This step in only used to determine the partition size, as explained at step 3.c.

b. With the deployment scheme wizard, create a new deployment scheme enabling redeployment.

c. For **Protected redeployment partition size**, give 200% of the size of the RAD file you have just created.

6. Preload the system profiles on your targets.

   a. Select the targets in the web interface.

   b. Select **Deploy now** in the contextual menu.

   c. Select **Redeployment preload** in the deployment wizard.

   d. Select the deployment scheme you have just created.

   e. Select the two OS configurations that you have prepared.

   f. Optionally, select additional software modules.

   g. Click **Customize GUI** if you want to customize the boot menu appearing on the target.

Scenario

You want to perform a multi-boot installation with three operating systems which have all unattended deployments.

Procedure

1. Create each unattended profile, and do not duplicate existing profiles. At creation time, define in the profile wizard the partitions where you must deploy the different operating systems. The partition must be the same for each profile. For example, if you deploy the first profile in the first partition, the second profile in the second partition and the third profile in the third partition you must create each profile as follows:

   - Create the first profile by specifying in the profile wizard three partitions and checking the first partition as the one on which the operating system should be installed.
   - Create the second profile by specifying in the profile wizard three partitions (same partition layout as in the first profile) and checking the second partition as the one on which the operating system should be installed.
   - Create the third profile by specifying in the profile wizard three partitions (same partition layout as in the first profile) and checking the third partition as the one on which the operating system should be installed.

   In this way, the first profile has the first partition marked as bootable, the second profile has the second partition marked as bootable, and the third profile has the third partition marked as bootable.

2. Edit each profile partition layout and set the **Must be deployed** and **Must be redeployed** options to **No** on the partitions where the operating system is not installed. Set the **Must be deployed** and **Must be redeployed** options to **Yes** on the partitions where the operating system is installed. The result should be the following in the three profiles:

Table 10. Partition options in the first profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	Yes	No	No
Must be redeployed	Yes	No	No

Table 11. Partition options in the second profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	No	Yes	No
Must be redeployed	No	Yes	No

Table 12. Partition options in the third profile

Option	Partition 1	Partition 2	Partition 3
Must be deployed	No	No	Yes
Must be redeployed	No	No	Yes

3. Export the profiles into a single rad file and set the size of the redeployment partition in the scheme as 200% of the rad size at least.
4. Deploy the profile configurations into a single deploy now operation on the target specifying the redeployment. Using the profile wizard, customize the GUI to have three options with the possibility to boot on each deployed operating system.

Configuring a deployment scheme for redeployment

Redeployment is a feature that affects *how* the target is being preinstalled, not *what* is in the deployed OS configuration. Redeployment is enabled by customizing a deployment scheme.

Because redeployment is basically the replay of a standard deployment operation, you must first configure a regular deployment process, and try it on a test computer. When you have performed these two stages, follow the instructions provided to turn your one-time deployment OS configuration into a redeployment OS configuration.

To customize a deployment scheme for redeployment, you can

- Create a new deployment scheme with the deployment Scheme Wizard
- Modify an existing deployment scheme with the deployment Wizard
- Edit the parameters of an existing deployment scheme manually

The following steps are based on the first and second options, which are very similar.

1. Follow the first alternative to create a completely new scheme, and the second alternative to modify an existing scheme with the wizard:
   - Go to the **Task templates** page and click **New deployment scheme**. This launches the deployment Scheme Wizard, which guides you through the customization of deployment parameters.
   - Go to the **Task templates** page. Select a deployment scheme, and click **Edit parameters using a wizard**.
2. Follow the instructions of the wizard in the same way as for a regular deployment, until you reach the panel called **On-site deployment features**.
3. Select **Enable support for quick redeployment of the same OS configuration** and click **Next**.
4. On the next panel, **Redeployment option**, select **Yes, keep** IBM Tivoli Provisioning Manager for Images **images in a protected partition**. Optionally modify the space that you want to allocate to this special partition, and click **Next**.

Note:

a. The protected partition size must be at least as large as the total size of all system and software images to be deployed on the computer, because it retains all these images. If you are unsure of the space required, start with approximately 800 MB for a Windows 2000 configuration, 1500 MB for a Windows XP configuration, or 1500 MB for a Linux configuration. If you want a more precise number, check the image sizes reported in a deployment log, and round up the total to accommodate the miscellaneous structures used for redeployment.

b. The space that you allocate to the redeployment partition is subtracted from the hard-disk total capacity detected by Windows or Linux. The user cannot detect, access, or delete this protected area from the operating system disk manager. It is not simply a hidden partition, but a hardware-protected area, as defined in ATA-5 specification. If necessary, you can recover this space by running another deployment operation.

5. Click **Finish** to complete the customization process and obtain a deployment scheme ready for redeployment.

Edit the parameters manually:

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select a deployment scheme
3. Click **View deployment parameters**
4. Click **Edit** in the section header in which you want to modify parameters.

Preloading for redeployment

Before you can redeploy a target, you must preload one or several OS configurations.

For a successful redeployment, targets must not **Boot on hard-disk if idle**. Make sure this target parameter is not selected for the targets you want to redeploy.

LVM partitions are not supported for redeployment. Make sure the OS configuration you want to redeploy does not contain an LVM partition.

After you have created an appropriate redeployment scheme, you can begin the preload of the OS configurations of your choice on the target. This operation must be initiated using the Target Monitor page of the web interface.

1. Select the targets to deploy and select **Deploy now** from the contextual menu to start the deployment wizard.
2. Select **Redeployment preload** and click **Next**.
3. Follow the instructions of the deployment wizard.

   Note:

   a. When you select a deployment scheme, only those configured for redeployment are displayed. If you do not have any scheme ready for redeployment, a warning message appears.

   b. Preloading more than one OS configuration is supported, but increases the preload time.

The preload automatically starts when the targets boot, just like with regular one-time deployments. The process goes through the same steps, with one exception. When Sysprep or `LinPrep` has completed and after all software modules have been installed, an image of the fully configured target is stored on the redeployment partition. If you have selected multiple OS configurations, the process repeats for all OS configurations in turn, until all redeployment images are ready.

Customizing the redeployment menu

You can customize the menu entries that you see in the user interface when starting a target in redeployment mode. Each OS configuration can define one or more menu entries, and the complete menu is the union of all entries defined by all available OS configurations.

After having selected **Redeployment preload** in the deployment wizard and selected the deployment objects:

1. Click **Customize GUI** in the deployment wizard. This opens the menu customization interface which is divided into three parts:
   - A left column with instructions on how to modify the menus and editable fields
   - A bottom banner with action buttons
   - A view of the target screen as it will appear
2. Click **New menu item**.
3. Modify the captions and actions.
4. You can select one of the following actions:
   - **Format and restore**
   - **Quick restore**
   - **Boot on OS**
5. If you want to protect a specific menu item from unauthorized users, you can set up a global password or user authentication for that user by selecting an appropriate value under **Authentication**. To make full use of this feature, you must first have defined authentication domains in the **Server parameters**. Three authentication formalisms are available

   Authenticate locally on RAD *group*
   uses the local user database to authenticate a user. The optional *group* parameter can be used to restrict the verification to a specific group of users. This type of domain is supported by both Windows NT and UNIX versions of the OS deployment server.

   Authenticate on NT server *server:group*
   forwards authentication requests to the NT server specified by the mandatory parameter *server*. The optional parameter *group* can be used to restrict the verification to a specific group of users. This type of domain is supported by the Windows implementation of the OS deployment server only.

   Authenticate on Radius server *ipaddr:secret*
   forwards authentication requests to the Radius-compliant device specified by the parameter *ipaddr*. The value of the parameter *secret* is used as the secret for the Radius communication, and must match the secret stored in the configuration of the Radius device for the protocol to work.

   Note: Authentication with redeployment does not work if the target is offline (the target has no network connection and boots from the hard disk). A message warns the user. If you plan to redeploy offline, use a global password rather than user authentication.
6. Click **Save** and then **Close** to exit this window.

Formatting hard disk and restoring files:

With this option, your partitions are always reformatted and all the files restored before you boot into the operating system.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network). If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

Note: Before redeploying Linux profiles, assign a label to the each profile partition or define a corresponding device name in the `/etc/fstab` file.

On the target, select the OS configuration to be restored.

After an OS configuration has been selected, Tivoli Provisioning Manager for Images completely format the disk and then restore all files. The default behavior is to:

1. Format the disk partitions as specified in the system profile.
2. Restore all the files from the hidden partition.
3. Boot on the selected operating system.

Using quick redeployment:

This option is the typical way to use redeployment. A fast verification of partitions and files is run and, fixes are performed if needed before the target boots into the operating system.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network). If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

On the target, select the OS configuration to be restored.

After an OS configuration has been selected, Tivoli Provisioning Manager for Images automatically restores it as quickly as possible. The default behavior (which typically takes only a few seconds to run) is to:

1. Verify that the disk partitions match the wanted system profile, and fix them if needed.
2. Verify that all partitions have the appropriate file content, and fix them if needed.
3. Boot on the selected operating system.

Booting on the installed operating system:

This option allows you to boot on the currently installed operating system, without any verification. It is fast, but it does not prevent operating system corruption.

After your targets are preinstalled for redeployment, they always boot into the user interface, independently of the selected boot order in the BIOS (disk or network).

If user authentication has been configured, targets connect to the OS deployment server using the PXE network adapter even if they start from the hard-disk.

On the target, select the option that allows you to boot on the operating system.

The target boots directly in the installed operating system, without any disk partition or file verification.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for Linux

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.

2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
3. Click **Edit** in the **General settings** section.
4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
   - **No**
   - **Whatever the notification is**
   - **If deployment is successfully completed**
   - **If deployment failed**
5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.

- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Bindings created during deployment

The Target Monitor creates a binding between the OS configuration chosen for the deployment and the targets being deployed. This binding is added into the database and can be later removed using the Target Monitor.

Because at least one configuration binding now exists, targets that have been deployed no longer show the locked screen. They show a boot menu with a list of the OS configurations that are bound to the target. This allows the target user to manually restart the deployment of an already deployed OS configuration by clicking on the corresponding line in the menu.

What you can do

You can remove, add, or modify OS configurations and software bindings using the Target Monitor.

Chapter 4. Provisioning VMWare ESX Server on x86 and x86-64 targets

This section provides information on how to work with the product to deploy VMWare ESX Server.

System profiles for VMWare operating systems

A system profile is the partition layout and list of files to deploy an operating system, either by unattended setup or by cloning, from a reference target or from a reference image file.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- When deploying an ESX 3.5 u5 system profile on HP servers with two RAID arrays, you must manually add the following commands in the `ks.cfg` configuration file:

  ```
  clearpart --all --drives=cciss/c0d1
  zerombr yes
  ```
- When deploying an ESX 4 system profile on an HP server with an HP P400 smart array disk controller, you must force the device mapping of the first disk to `/dev/cciss/c0d0`.
- When deploying an ESX 4 system profile on an HP server with an HP P410 smart array disk controller, you must force the device mapping of the first disk to`/dev/sda` or `/dev/sdb`.
- Disk cloning is not supported for ESX systems. Only unattended setup is supported.
- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.
- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.

- The exFAT filesystem is not supported.
- Before deploying a system profile to a target ensure that the root partition is C.
- You cannot deploy Linux profile with an LVM root partition if you use deployment media.

Creating an unattended setup system profile for VMWare

VMWare unattended system profiles must be created on a target, running the web interface extension. It can either be the OS deployment server itself, or a remote target whose IP address must be entered in the profile wizard.

To create an unattended setup system profile for VMWare ESX 3.5, you must download the binary file named *ESX Server 3.5 Update 2 CD image* (596 MB). Creating the profile from *ESX Server 3i U2 Installable* (238 MB) results in a failed deployment.

To create a new system profile:

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select your operating system from the list and click **Next**.
5. Follow the instruction of the profile wizard.

On VMware ESX 4.0, the service console runs in a virtual machine. When you view the partitions of your system profile, this virtual machine is shown as a logical disk named *esx console*, which is not mapped to any physical device. You can therefore deploy the profile on a target with only one disk. Do not edit the *esx console* logical disk.

When your first unattended installation profile is created, you can use it to deploy targets.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles

To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.

2. Click **Edit** on top of the parameter sections to edit the parameters.

Browsing partition files

You can browse partition images stored on your server.

1. Go to **Server** > **OS deployment** > **System profiles**. Double-click on a profile to view the details.
2. In the **Original partition layout** section, click **Browse image of primary partition 1**.
3. You can expand or update the whole partition or a part of it.
   - To expand the whole or part of the partition:
     a. Right-click the folder you want and select **Expand on local disk**.
     b. Choose the computer where you want to expand and store the files contained in the selected partition.
     c. Specify the destination folder where to extract the partition files.
     d. Follow the instructions of the image wizard to expand the partition.

        Note: You must expand the partition to an empty directory. If you select a folder that is not empty the extraction fails.
   - To update the whole or part of the partition:
     a. Right-click the folder you want and select **Update from local disk**.
     b. Specify the source folder of the OS deployment server where your updated data are located.
     c. Specify the destination folder where to extract the partition files.
     d. Follow the instructions of the image wizard to update the partition.

On the **Partition image explorer** page, you can create a new directory by selecting **Add new directory** in the contextual menu. You can also modify or add files by selecting **Upload file** in the contextual menu.

Note: File upload is limited to 16 MB.

Changing the partition layout

You can update partition layout to resize partitions, assign mount points, or change the file system.

If you change the partition layout in system profiles, you might render the profile unusable. It is recommended not to change the partition layout in system profiles, unless you know that the changes you want to make have no side effect.

In any case, do not:

- Add or remove a partition before the *root* partition.
- Transform a primary partition into a logical partition.
- Move a primary partition before *boot* or *root* to LVM.

Note: Changing the partition layout from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose either one or the other entry point, and then perform all your changes from that entry point.

By editing the partition layout, you can:

- Add or delete partitions.

  Note: Adding or deleting partitions can lead to OS configuration problems. Use this feature carefully. To provide a clear description to your profile, use the **Comment** field.

- Resize a partition by dragging sliders, or by assigning it an absolute or relative size.
- Change the file system of a partition.
- Assign a mount point to the partition.

Note: When in the disk partitioning, both fixed sizes and percentage sizes are included, the exact reproduction is not guaranteed.

1. Click **Edit partition layout** on either the **Profile details** page or the **OS configuration details** page, **Disks** tab.
2.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

        Linux In a Linux profile, assign a mount point for the new partition. To be valid, this mount point must reference an existing directory in the main image. Starting from Fix Pack 3, the Linux profiles with the root partition as LVM are supported. In this case, you must ensure that the HTTP mode is selected in the deployment scheme when deploying the profile. With the root partition as LVM, you cannot perform the deployment using the media.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.
3.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

        Windows In a Windows profile, the operating system deployed using a system profile must be installed on the C: drive. Other drive letters are not allowed for the bootable partition.
   - To resize partitions with the sliders, drag the slider to the right of the partition.
   - To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.
4.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.
     d. Indicate the partition properties, including a mount point and click **OK**.

- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

Modified partitions are aligned on megabytes rather than on cylinders. The following warning message might appear in the logs and can be safely discarded: `Warning: partition x does not end at a cylinder boundary`. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

If you want to use the same system profile with two different partition schemes, you can also duplicate a system profile by right-clicking the profile name and selecting **Duplicate profile**. The copy shares the same image files, but can have a different partition layout.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.
3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

   Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/cxdx`, where *x* is the disk number. Edit your device mappings accordingly.
4. Click **OK**.
5. Repeat step 2 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the very bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies distinct.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Note: For VMWare ESX platforms, it is impossible to set the host name of the operating system if the network configuration is dynamic (DHCP). Setting up the target host name is only supported with static TCP/IP settings.

Editing OS configuration parameters in the web interface for VMWare

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Editing a VMWare parameter file

You can modify OS configuration parameters by editing a file. This option allows you to modify parameters that are not displayed in the web interface. However, you must be experienced to use this option advantageously, because Tivoli Provisioning Manager for Images does not provide any syntax checking of the file. Information about the file format and syntax can be found in the documentation of the operating system itself.

1. Click **Edit custom 'ks.cfg'** to edit the file to modify the size of the VMFS and VMKcore partitions if needed and to define a custom partitioning scheme when installing VMWare with scripted installation.

   Note: Information about partitions in the `ks.cfg` custom configuration file is not normally taken into account.
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). Unless otherwise specified, parameters specified in the default file override the content of the custom file.

In the following example, the following partitions are created:

- A `ext3` partition of 900 KB on the `sda` disk.
- A `vmfs3` partition of 50 MB is created on the `sda` disk.

- A vmkcore partition of 94 KB on the sda disk.

```
part /var --fstype ext3 --size=900 --ondisk sda
part None --fstype vmfs3 --size=50000 --grow --ondisk sda
part None --fstype vmkcore --size=94 --ondisk sda
```

Troubleshooting:

If the OS configurations in the deployed operating system are not what you expected, you must examine carefully the parameter files. They are the result of the merge between the custom file and the default file created. See the log file Windows/Panther/unattendGC/setupact.log for problems in the file merge.

Note: Ensure you specify the full paths for the commands you use in the unattend.xml file.

To troubleshoot OS configuration parameters after a failed deployment, see the /tmp/anaconda.log file.

Task templates for VMWare operating systems

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout
The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout
The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used

when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout
The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout
The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Customizing a screen layout

You can customize the screen layout of a target.

To customize a screen layout:

1. Select the layout that you want to customize in the right pane of the **Task Templates** page of the web interface page.

   Note: An actual layout must be selected and not a layout folder (left pane)
2. At the bottom of the page, the screen layout is shown in reduced size. Click **Customize GUI** to open the screen layout editor.
3. The editor is composed of a left column, containing instructions, a *What-You-See-Is-What-You-Get* (WYSIWYG) view of the screen being edited and a bottom banner with action buttons.
4. Click on the action buttons or directly on the items that you want to modify to see their editable properties displayed in the left column. Make the wanted changes and then click **Save** to keep your new screen design. Return to the **Task Templates** page by clicking **Back**.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard. Go to **Server** > **OS deployment** > **Task templates**. and click **New deployment scheme**.

2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section. Some advanced deployment scheme features are available only in this mode and not through the wizard.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   The following parameters apply for simple one-time deployments and for redeployment operations.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, so the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters
   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

   Edit parameters for targets that are not yet in the database
   to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

   Never edit parameters
   to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

   You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

   No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

   Yes, display a warning
   if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment

if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, because Tivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of targets to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed target, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, / on UNIX operating systems or `c:/` on Windows operating systems. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: Multicast is available only if:

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of targets that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast

to deploy targets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast

- **without explicit synchronization**

  to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.
- **with synchronization on all targets**: to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.
- **with the following synchronization parameters:**

  to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

In addition to that, you can also configure two additional parameters:

- `Multicast block size (MB)` with default value 16
- `Multicast TOC block size (KB)` with default value 32

During a deployment configured to use multicast, the server splits the system profiles in blocks that are sent to all the targets in the same multicast group. Before sending each block, the server sends a packet, called ToC, that describes the content of a block. The parameter `Multicast block size` is the maximum size of a block of data. The second parameter `Multicast TOC block size` is the maximum size of the ToC.

These parameters can be configured to improve the performance of the multicast deployment. If you have a high speed network rated at 1 GB and a profile of 15 or 20 GB, it is recommended to change both the values to at least 64.

For Windows Vista and later, and Linux platforms, you can decide to use a network share on Windows, or Linux HTTP, to download the files to the targets , rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP provides a shorter installation time. For Linux deployments, you can also allow the use of the Linux Deployment Engine for 64-bit cloned profiles using PXE.

To use a network share on Windows or Linux HTTP:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter in the **Network share module** section. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module).

On Linux only, to use the Linux Deployment Engine:

- Select **Use Linux Deployment Engine x86-64 if applicable** to deploy a Linux cloned 64-bit profile using PXE, for driver injection purposes. This option is ignored in all other cases.

On-site deployment

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment option

Indicate if you want to keep the deployment image in a protected partition and the size of this partition. These options are valid only to configure the deployment scheme for redeployment. More information is available in deploy/tosd_redeplscheme.dita.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation

This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

This parameter, located in the **General settings** section, is set to **No** by default. Setting this parameter to **Yes** unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user. Review the section on network boot scenario of the deployment process topic.

Unbind software module at the end

This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is relevant for Windows Vista/2008/7/2012 deployments only.

Send mail at end

This option can be set only if a *sendmail* tunnel has been created.

Send mail to

This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network

This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button

This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Creating media for deployment for VMWare

You can create deployment media such as CDs, DVDs, or USB drives to install machines without connecting them to the OS deployment server.

You can use this kind of deployment when there is no connection or connection to the OS deployment server is very slow.

Some typical situations are small branch offices with slow links and no local deployment server, isolated computers with no connection to an internal network, laptop users currently away from LAN or connected using a modem.

If the data you want to use does not fit on a single CD or DVD, use a USB drive.

You can also create offline Win PE-based deployment media in kernel-free mode. You select the deployment engine type and the architecture to use, as well as System Profiles and any software module you want to include in the deployment media. If you have more than one WinPE deployment engine, you can select the target models and the matching engines for these models are included. The engines are cached locally on the target.

Note:

- You must create the deployment media from an OS deployment server or a web interface extension installed on a computer with the same byte order (little endian or big endian) as the one on which you want to use the deployment media.
- To deploy Windows system profiles on Hyper-V, make sure that the boot order indicates the hard drive before the CD-ROM or USB drive.
- Redeployment is not available when deploying from a deployment media.

Creating an OS deployment USB drive with the wizard

Tivoli Provisioning Manager for Images can automatically generate deployment USB drives that replay the deployment process for a given system profile or for any kind of software modules available.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must be formatted as FAT32 or NTFS.

Note: SuSE Linux Enterprise Desktop cloning is not supported on USB drive deployments.

Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The deployment USB drive is self-contained and can be used instead of a CD or DVD to provision a target entirely offline, without using the OS deployment server. These deployment USB drives can also be used to deploy computers without a PXE-compliant network adapter.

To create OS deployment USB drives:

1. Perform one of the following operations:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.
   - Go to **Server** > **OS deployment** > **Software modules**.
2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment USB key** to start the USB key wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. (*Optional*) Change settings for targets running the USB key that you are creating.

   Included objects

   When selecting objects to be included, be aware that:
   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile and exactly one deployment scheme must be included in your image.
   - The software application order is automatically included.
7. If your USB key has already been used as a deployment media, you might choose to keep a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB key.
8. Plug your USB key into a machine running the web interface extensionand specify its address.
9. Choose the drive matching your USB key.
10. Click **Finish** to close the wizard.

Use the USB drive to deploy a given system profile or any kind of software module.

Creating an OS deployment USB drive with command lines

You can create an OS deployment USB drive that Tivoli Provisioning Manager for Images can use when a target cannot boot from the network.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must have boot capabilities and a FAT32 or NTFS filesystem. The drive must be already formatted; existing files on the partition are not deleted. USB keys already filled with a bootable operating system might not work.

Note: Refer to the *Troubleshooting and support* set of topics for information about problems or limitations related to deployments using a network boot USB drive, and to the product release notes or the readme file provided with the fix pack for the most up-to-date information related to problems or limitations.

The command line must be used only when the web interface is either inappropriate or unavailable.

Use this command line:

- On Windows operating systems:

```
rbagent.exe -s <OSD_server_ip_address>:<OSD_server_password>
rad-usbget <drive>
keepshared|delshared preferwpe|prefermcp nodes
```

Where:

OSD_server_ip_address
Is the IP address of the OS deployment server.

OSD_server_password
Is the password for the administrative user (typically admin) on your OS deployment server.

drive
Is a drive letter of the Windows target where you run the rbagent command. The rad-usbget command adds requested files to the FAT32 or NTFS partition and makes it bootable. The drive must be already formatted. Existing files on the partition are not deleted.

keepshared
Keeps a shared repository of previous data to improve data regeneration speed. If you keep the existing shared repository, you might use more space on the USB drive.

delshared
Deletes a shared repository of previous data.

preferwpe|prefermcp
Defines if the Linux deployment engine environment or WinPE is loaded from the USB drive, when a target boots from this USB drive, without accessing the network. Only when Linux deployment engine or WinPE is running, does the target connect to the network and try to contact an OS deployment server. If you deploy only Linux, specify prefermcp to skip WinPE. You can specify preferwpe only if there is a WinPE deployment engine on the OS deployment server.

nodes
Defines the deployment settings with a space-separated list of objects. Specify at least DEPLSET:Default for the deployment schema, and PROFILE:SystemID for the system profile.

You can now boot the target using the OS deployment USB drive instead of the network card. To use the PXE emulation USB key, insert the USB key into the drive and restart the target. If your machine does not boot from the USB key, check the BIOS boot list to see if your optical drive is included in the boot sequence and is listed before the hard disk. Most machines also allow you to select the temporary boot device without changing the boot sequence in BIOS.

Creating OS deployment CD and DVD

Tivoli Provisioning Manager for Images can automatically generate deployment CDs and DVDs that replay the deployment process for a given system profile or for any kind of software modules available. You can use this feature to create OS deployment CDs and DVDs that can be easily sent through the Internet or by e-mail, to refresh a computer back to its initial working state after installation.

The CD/DVD deployment occurs in kernel mode. Microsoft tools are used to build the CD/DVD. By specifying the target models, the product automatically determines which deployment engine to use and the drivers corresponding to the specified target models are added to the CD/DVD. These CDs and DVDs can also be used to deploy computers without PXE compliant network adapter. The creation of DVDs and media spanning is supported. These media can be protected using an activation code preventing unauthorized personnel from using it.

To create OS deployment CD and DVD:

1. Perform one of the following operations:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.
   - Go to **Server** > **OS deployment** > **Software modules**.
2. Click **Generate Media** or select **Create deployment media** in the contextual menu.
3. Select **Create a deployment CD or DVD** to start the CD and DVD wizard. Click **Next**.
4. Specify the operating system for which to build the CD or DVD. Select **Windows** to load a WinPE deployment engine, **Linux** to load a Linux deployment engine environment, or **Both** to load both.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. Follow the wizard instructions to create an ISO image.

   Included objects

   When selecting objects to be included in the ISO image, be aware that:
   - The wizard displays all the deployment schemes, system profiles, and software modules currently stored on your OS deployment server.
   - At least one system profile and exactly one deployment scheme must be included in you image.
   - The software application order is automatically included.

   Hardware options

   In the hardware options settings some boot options can be customized. By default the options are unchecked but some special cases can require changes. In particular, if the CD or DVD is to be used on a USB drive or as a secondary drive, it might be necessary to specify the option **use BIOS for CD or DVD ROM access**. When this option is selected, on some hardware it might also be necessary to select **disable enhanced disk access** (for IDE CD or DVD) or **disable USB** (for USB CD or DVD) to ensure that Tivoli Provisioning Manager for Images use of other IDE or USB devices does not interfere with the BIOS access to the CD or DVD. In addition, deploying from the second CD or DVD drive of a target only works if you can ensure that subsequent boots keeps booting on the same CD or DVD drive.

   Security issues

   For security issues, you might want to protect deployment from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment to proceed.

You might also want to hide the content of the ISO image that contains sensitive information such as product keys. To do this, select **Hide the content of CD or DVD** in the CD or DVD Wizard. If you then try to access files in your ISO image, you see the content as `CDROM_content_hidden`.

Size of the ISO file

The wizard allows you to choose the size of the ISO images.

a. Enter the maximum size in the field displayed.

b. Click **Next** and the wizard starts to precompute the ISO file size.

The wizard displays the results for the number of disk images and the size required. You then have the option to:

- Download it directly from the server.
- Use the web interface extension
- Generate it on the server itself in the import directory.
- Generate it on another computer running the web interface extension

Note:

- When creating the ISO files, all objects of type *single file to copy*, *image headers*, and *WIM images* (which includes Windows Vista/2008/7/2012 unattended setup profiles), are put on the first CD or DVD. Therefore, the first ISO file might grow larger than the requested spanning size if the total size of the files to be put on the first ISO requires it.

  For example, if you try to create an OS deployment DVD containing both Windows Vista/2008/7/2012 unattended setup profiles, both profiles must be contained on the first ISO, but their total size is larger than 4 GB. Therefore, the ISO cannot be burned into a single layer DVD. In this case, either use a double layer DVD, or transfer the ISO without burning it.
- When deciding where to generate the ISO image, be aware that:
  - If the estimated size is bigger that 2 GB, do not use the link to download directly from the server, because of limitations of web browsers. An exception to this rule is Mozilla Firefox on Linux, which can extract files as large as 4 GB or more.
  - Because of file system limitations, do not extract files bigger than 4 GB on FAT32 partitions.

Use a CD creation tool to burn the ISO image onto disks.

Note: Vista 2008 Windows 7 Windows Vista/2008/7/2012 unattended setup profiles contain at least one file larger than 1 GB which cannot be split. Therefore, ISO files containing Windows Vista/2008/7/2012 unattended setup profiles must be burned on a DVD.

If you encounter problems when deploying from this CD or DVD on a virtual machine, make sure that the CD drive comes after the hard disk in the boot order.

Setting up an activation code

For security issues, you might want to protect deployment or booting from the CD with an activation code. When your computer boots on the CD, the activation code is required for the deployment or the network boot to proceed.

To prevent being asked several times for the activation code during deployment:

- The deployment scheme included on your deployment CD must have the network setting **Use 'BIOS fall back MBR' to start PXE** set to **No**.
- The boot order of your target must be set to hard disk first and you must boot on the CD manually the first time.
- To set up an activation code for the first time, when creating the deployment CD:
  1. Select **Include activation code protection** in the deployment media wizard.
  2. Enter and confirm the chosen password. You must remember this password if you want to obtain other activation codes for this CD.
  3. Set a password expiration date under **Valid until**.
- To obtain a new activation code, for example, if you must use the CD after the current activation code expiration date:
  1. Click **Generate Media** on the Profiles page to start the deployment media wizard.
  2. Select **Generate a new activation code**.
  3. Click **Next** and follow the wizard instructions to obtain your new activation code. You must remember the password given when creating the first activation code for this CD.

The wizard provides you with the generated activation code that you need when using the CD.

Deploying VMWare

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

The deployment process

In Tivoli Provisioning Manager for Images, a deployment is made up of several steps that are automatically run in sequence without user interaction:

Deployment process

1. Hardware configurations are potentially deployed, for example to create RAID volumes.
2. Partitions are created on the hard-disk, and then formatted according to information contained in the System profile.
3. All deployment objects (system profiles partition files, and software modules) are downloaded to a temporary storage location on the hard-disk.
4. Operating system files are written in the hard-disk partitions, creating a bootable operating system with files and applications configured by database bindings between the *target* and *software modules*.
5. Target specific configuration, such as the *host name* or the *product key* are gathered from the database to create a textual configuration file used by the system preparation tool.
6. The operating system is started, allowing Sysprep or LinPrep to configure the operating system according to information stored in the Tivoli Provisioning Manager for Images database.
7. Additional software are optionally installed, if they need to be installed after the operating system.
8. The temporary storage location is cleaned. Installation files are removed.

9. Tivoli Provisioning Manager for Images takes control again when Sysprep or LinPrep has completed and rebooted the target, and displays a message indicating the status of the deployment.

When the deployment is complete, the operating system is installed and ready to be used by the end-user defined for this target in the database.

HTTP deployment

From version 7.1.1.3 of the product, the deployment of Linux system profiles in unicast can be performed through HTTP. By default, Linux setup deployment is performed using HTTP. Booting the Linux kernel, the kernel downloads the packages using HTTP. This results in a faster deployment and no third party is involved. In this way you also avoid that unallocated space is left on the target disk equal in size to the rembo cache used by the deployment. In non-HTTP deployment this space is not added to the partition defined as the 100% of the remaining disk space.

To activate or deactivate HTTP downloads in unicast, change the value of **Download files with a network share or Linux HTTP when applicable** in the **Network settings** section of your deployment scheme. By default, the value is set to **Yes**.

The deployment of Linux system profiles in multicast remains unchanged.

Network boot scenarios

Depending on the number of OS configurations bound to a specific target, a target behaves differently when it boots on the network:

- If no OS configuration is bound to the target (for example, when a target starts for the first time and has not been configured), a special screen is displayed that asks the administrator to configure an OS configuration binding for this target on the OS deployment server. Deployment is not possible until an OS configuration is bound to the target.
- If one or more OS configurations is bound to this target, but no deployment has been scheduled on the server, a screen is displayed with a list of all the OS configurations bound to the target. Clicking on an item in the list starts an interactive deployment for the selected OS configuration, using either the **Default** deployment scheme (if no deployment scheme has been configured for this target), or the deployment scheme used during the last deployment.
- If one or more OS configurations are bound to this target, and a deployment has been scheduled on the server for a specific OS configuration, the target immediately starts the deployment without requiring any user intervention.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.

- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

VMWare ESX 4.X

> The DHCP server must be configured to give a new IP address to the target on which you deploy the VMWare ESX 4.0 system profile, different from the one used for network booting. This second IP address is used by the console virtual machine within the hypervisor. You can use an open DHCP server, or a DHCP server with a range of *free* IP addresses in the pool, for example.
>
> It is not possible to use reservations, as the MAC address of the console virtual machine cannot be known in advance.

When you deploy a cloning system profile, you may encounter driver and/or disk mapping issues if the hardware of the reference target of the cloning system profile is significantly different from the hardware of the deployment target.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. Select **Simple deployment** and click **Next**
5. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Note: The Wake on LAN options are available only on Intel targets.

During the deployment of VMWare ESX 4.0, soon after the first boot into the ESX operating system, errors may show on the screen of the target.

```
[ERROR] open: no such file or directory
[ERROR] rtkh_open: cannot open /etc/rbotmp/rbagent.trc
[ERROR] rtkh_open: cannot open /etc/rbotmp/rbagent.log
```

These errors are normal and can safely be ignored.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date

- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the bom file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The bom file contains target-specific parameters for the given task.

The log file, the task file, and the bom file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for VMWare

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     - **No**
     - **Whatever the notification is**
     - **If deployment is successfully completed**
     - **If deployment failed**
  5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.
- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Bindings created during deployment

The Target Monitor creates a binding between the OS configuration chosen for the deployment and the targets being deployed. This binding is added into the database and can be later removed using the Target Monitor.

Because at least one configuration binding now exists, targets that have been deployed no longer show the locked screen. They show a boot menu with a list of the OS configurations that are bound to the target. This allows the target user to manually restart the deployment of an already deployed OS configuration by clicking on the corresponding line in the menu.

What you can do

You can remove, add, or modify OS configurations and software bindings using the Target Monitor.

Chapter 5. Provisioning VMWare ESXi on x86-64 targets

This section provides information on how to work with the product to deploy VMWare ESXi.

VMWare ESXi is supported from version 4.1.Version 5.0 and 5.1 have been tested and are also supported.

Note: Previous version of ESXi (3.5 and earlier) have never been supported.

System profiles for ESXi systems

A system profile is the partition layout and list of files to deploy an operating system.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.
- The exFAT filesystem is not supported.

Creating an unattended setup system profile for VMWare ESXi

VMWare ESXi is supported from version 4.1.Version 5.0 and 5.1 have been tested and are also supported (previous versions of ESXi, 3.5 and earlier, have never been supported). VMWare ESXi unattended system profiles must be created either on a Linux or a Windows target. The target must be running the web interface extension.

It can either be the OS deployment server itself, or a remote target whose IP address must be entered in the profile wizard.

To create a new system profile:

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select your operating system from the list and click **Next**.
5. Follow the instruction of the profile wizard.

When your first unattended installation profile is created, you can use it to deploy targets.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles

To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Changing the partition layout

Partition layout cannot be updated to resize partitions, assign mount points, change the file system.

The partition editor is disabled for VMWare ESXi system profiles, because it is impossible to modify the default partition scheme of the VMWare ESXi installer.

To add a new disk during a VMWare ESXi deployment, you must edit the custom `ks.cfg` file in the operating system configuration details of the profile. For more details about how to edit the custom `ks.cfg` file, see “Editing a VMWare ESXi parameter file” on page 219.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the very bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies distinct.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Note: For VMWare ESXi platforms, it is impossible to set the host name of the operating system if the network configuration is dynamic (DHCP). Setting up the target host name is only supported with static TCP/IP settings.

Editing OS configuration parameters in the web interface for VMWare ESXi

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Editing a VMWare ESXi parameter file

To add a new disk during a VMWare ESXi deployment, you must edit the custom `ks.cfg` file in the operating system configuration details of the profile.

1. Click **Edit custom 'ks.cfg'** to add a partition as in the following example. If you want to add a partition named `datastore2`, on the disk named `naa.5000c50007bd5c73`, with a default size of 20 GB, add into the custom `ks.cfg` file the following line:

   ```
   part datastore2 --fstype=vmfs3 --size=20000 --grow
   --ondisk="naa.5000c50007bd5c73"
   ```

   Note: It is possible to use keyword substitution in the custom `ks.cfg` file. In the following example, the User Category0 variable, defined at the configuration or target level, is used to replace the disk name:

   ```
   part datastore2 --fstype=vmfs3 --size=20000
   --grow --ondisk="{$User.UserCateg0$}"
   ```
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). Unless otherwise specified, parameters specified in the default file override the content of the custom file.

Troubleshooting:

If the OS configurations in the deployed operating system are not what you expected, you must examine the parameter files carefully. They are the result of the merge between the custom file and the default file created.

Task templates

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes

Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout

The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout

The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout

The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout

The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout

The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured

operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard. Go to **Server** > **OS deployment** > **Task templates**. and click **New deployment scheme**.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section. Some advanced deployment scheme features are available only in this mode and not through the wizard.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, so the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters
   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

   Edit parameters for targets that are not yet in the database
   to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

Never edit parameters
to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

Yes, display a warning
if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment
if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, because Tivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of targets to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed target, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, `/` on UNIX operating systems or `c:/` on Windows operating systems. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: For VMWare ESXi deployment, downloading files by HTTP is a requirement. Moreover, HTTP deployment is only available in unicast mode.

- Select **Unicast (no bandwidth sharing)**.
- Select **Download files with a network share or Linux HTTP when applicable**.

On-site deployment

Redeployment is not available for VMWare ESXi. Leave the check box blank and click **Next**.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Unbind OS configuration at the end

This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user (see Network boot scenarios in “The deployment process” on page 224.

Unbind software module at the end

This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is relevant for Windows Vista/2008/7/2012 deployments only.

Configure Network

This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button

This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Deploying VMWare ESXi

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

The deployment process

In Tivoli Provisioning Manager for Images, a deployment is made up of several steps that are automatically run in sequence without user interaction:

VMWare ESXi deployment is performed using the kernel-free flow and downloading files by HTTP.

1. The installer is downloaded from the OS deployment server by TFTP.
2. The installer is started.
3. The installer downloads by HTTP all the installation files stored on the OS deployment server.
4. The installer installs VMWare ESXi on the target.
5. The OS deployment server takes control again to display the result of the deployment.

When the deployment is complete, VMWare ESXi is installed and ready to be used.

Network boot scenarios

Depending on the number of OS configurations bound to a specific target, a target behaves differently when it boots on the network:

- If no OS configuration is bound to the target (for example, when a target starts for the first time and has not been configured), a special screen is displayed that asks the administrator to configure an OS configuration binding for this target on the OS deployment server. Deployment is not possible until an OS configuration is bound to the target.
- If one or more OS configurations is bound to this target, but no deployment has been scheduled on the server, a screen is displayed with a list of all the OS configurations bound to the target. Clicking on an item in the list starts an interactive deployment for the selected OS configuration, using either the **Default** deployment scheme (if no deployment scheme has been configured for this target), or the deployment scheme used during the last deployment.
- If one or more OS configurations are bound to this target, and a deployment has been scheduled on the server for a specific OS configuration, the target immediately starts the deployment without requiring any user intervention.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

Note:

- To deploy VMWare ESXi, you must use the kernel-free flow and download files through HTTP.
- Make sure the target on which you want to deploy your system profile is compatible with VMWare ESXi. VMWare provides a tool to check hardware compatibility. At the time of writing, this tool is available at http://www.vmware.com/resources/compatibility/search.php.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations. These bindings can be configured manually or with binding rules .

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. Select **Simple deployment** and click **Next**
5. Follow the deployment wizard instructions to select a deployment scheme and an OS configuration, and to set up deployment options.

   Note: The Wake on LAN options are available only on Intel targets.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

For VMWare ESXi 4.1, 5.0 and 5.1 a log file might not exist, while a task file and a `bom` file are always available.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for VMWare ESXi

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State
Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     - **No**
     - **Whatever the notification is**
     - **If deployment is successfully completed**
     - **If deployment failed**
  5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.

- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Bindings created during deployment

The Target Monitor creates a binding between the OS configuration chosen for the deployment and the targets being deployed. This binding is added into the database and can be later removed using the Target Monitor.

Because at least one configuration binding now exists, targets that have been deployed no longer show the locked screen. They show a boot menu with a list of the OS configurations that are bound to the target. This allows the target user to manually restart the deployment of an already deployed OS configuration by clicking on the corresponding line in the menu.

What you can do

You can remove, add, or modify OS configurations and software bindings using the Target Monitor.

Chapter 6. Provisioning non x86 and non x86-64 targets

This section provides information on how to provision targets which do not follow an x86 or an x86-64 architecture.

Provisioning Linux on PowerPC and Cell targets

To work with Linux system profiles on PowerPC and Cell targets, you must take into account some specificities of these targets.

DHCP specificities

There are specific considerations for setting Dynamic Host Configuration Protocol (DHCP) options. Make sure you set them appropriately.

Note: Microsoft DHCP server does not work well with somePowerPC firmware. Use IBM recommended DHCP servers.

Registering new targets

You must add targets manually into the Target Monitor or import a comma-separated text file containing a list of targets to be added.

PReP boot and /boot partitions

The PReP boot partition is mandatory to deploy Linux on PowerPC. It must be the first partition of the disk. If your profile contains a /boot partition, this partition must be the second partition. Set both partitions to a fixed size in MB and not in percentage of the total disk size.

System profiles on Linux operating systems on PowerPC

A system profile is the partition layout and list of files to deploy an operating system, either by unattended setup or by cloning, from a reference target or from a reference image file.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- Disk cloning is not supported for Linux PowerPC and Cell targets. Only unattended setup is supported.
- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.
- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.
- You cannot deploy Linux profile with an LVM root partition if you use deployment media.

Creating an unattended setup system profile for Linux on PowerPC

SUSE

SuSE Linux Enterprise Server (SLES) 10 unattended system profiles for PowerPC must be created on a Linux target, running the web interface extension. It can either be the OS deployment server itself, or a remote target which IP address must be entered in the profile wizard.

You can install operating systems using standard installation processes in unattended mode.

To create a new system profile:

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select your operating system from the list and click **Next**.
5. Follow the instruction of the profile wizard.

When your first unattended installation profile is created, you can use it to deploy computers. The profile wizard for a Linux unattended installation helps you to create a partition layout for this profile. Mandatory partitions are:

- **PReP Boot**
- **Boot**
- **Swap**
- **Root**

The **PReP Boot** partition has a size of 256 MB, the **Boot** partition of 100 MB. **Swap** and **Root** partition sizes are editable. The suggested settings in the profile wizard should be kept if there is any doubt in the allocation of disk space.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.

2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles:
To display and edit the parameters associated with a given profile:
1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Changing the partition layout

You can update partition layout to resize partitions, assign mount points, or change the file system.

If you change the partition layout in system profiles, you might render the profile unusable. It is recommended not to change the partition layout in system profiles, unless you know that the changes you want to make have no side effect.

In any case, do not:
- Add or remove a partition before the *root* partition.
- Transform a primary partition into a logical partition.
- Move a primary partition before *boot* or *root* to LVM.

Note: Changing the partition layout from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose either one or the other entry point, and then perform all your changes from that entry point.

By editing the partition layout, you can:
- Add or delete partitions.

  Note: Adding or deleting partitions can lead to OS configuration problems. Use this feature carefully. To provide a clear description to your profile, use the **Comment** field.
- Resize a partition by dragging sliders, or by assigning it an absolute or relative size.
- Change the file system of a partition.
- Assign a mount point to the partition.

Note: When in the disk partitioning, both fixed sizes and percentage sizes are included, the exact reproduction is not guaranteed.
1. Click **Edit partition layout** on either the **Profile details** page or the **OS configuration details** page, **Disks** tab.
2.
   - To add a partition:
     a. Click **Modify partition layout**.
     b. Click into an existing partition.
     c. Click **Add a partition** in the contextual menu.

    d. Indicate the partition properties, including a mount point and click **OK**.

       Linux In a Linux profile, assign a mount point for the new partition. To be valid, this mount point must reference an existing directory in the main image. Starting from Fix Pack 3, the Linux profiles with the root partition as LVM are supported. In this case, you must ensure that the HTTP mode is selected in the deployment scheme when deploying the profile. With the root partition as LVM, you cannot perform the deployment using the media.

- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

3.

- To add a partition:
    a. Click **Modify partition layout**.
    b. Click into an existing partition.
    c. Click **Add a partition** in the contextual menu.
    d. Indicate the partition properties, including a mount point and click **OK**.

       Windows In a Windows profile, the operating system deployed using a system profile must be installed on the C: drive. Other drive letters are not allowed for the bootable partition.
- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

4.

- To add a partition:
    a. Click **Modify partition layout**.
    b. Click into an existing partition.
    c. Click **Add a partition** in the contextual menu.
    d. Indicate the partition properties, including a mount point and click **OK**.
- To resize partitions with the sliders, drag the slider to the right of the partition.
- To update all other parameters, select a partition by clicking it, and select **Edit partition** in the contextual menu.

Modified partitions are aligned on megabytes rather than on cylinders. The following warning message might appear in the logs and can be safely discarded: `Warning: partition x does not end at a cylinder boundary`. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

If you want to use the same system profile with two different partition schemes, you can also duplicate a system profile by right-clicking the profile name and selecting **Duplicate profile**. The copy shares the same image files, but can have a different partition layout.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.
3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

   Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/cxdx`, where *x* is the disk number. Edit your device mappings accordingly.
4. Click **OK**.
5. Repeat step 2 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies unique.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Editing OS configuration parameters in the web interface for Linux on PowerPC:

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.

2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Editing a Linux parameter file for Linux on PowerPC:

Note: Starting with version 7.1.1, information about partitions in the custom configuration file is not normally taken into account.

SUSE For partitioning information in the custom configuration file to be taken into account, and to replace any information in the default file, these conditions must be fulfilled:

- The version of the product must be 7.1.1.3 or higher
- The deployment must be performed by HTTP
- The system profile must be of type *unattended setup*
- The operating system being deployed must be SuSE

Information in **Common networking info** is overwritten by the information in the custom configuration file. However, information in the **Advanced network settings** is not overwritten because it is applied post-configuration.

1. On **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**:
   - Red Hat To edit the file, click **Edit custom 'ks.cfg'**.

     Note: If you are deploying Linux on machines with two disks, add one of the following statements to the `ks.cfg` file:

     ```
     bootloader --driveorder=sdb,sda
     ```

     or

     ```
     bootloader --driveorder=hdb,hda
     ```

     depending on the disk naming system of the machines.
   - SUSE To edit the file, click **Edit custom 'autoinst.xml'**.

     You can use the following sections in your file:
     - `<files>`
     - `<groups>`
     - `<users>`
     - `<signature-handling>`
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). The resulting configuration is the union of the values in the custom and default files, with the following restrictions:

- The result of conflicting values between the custom and default files is undefined.

- Partition information in the custom file is taken into account only for SuSE unattended setup by HTTP, in which case only the information in the custom file is taken into account.
- Advanced network settings are always applied, because they are performed at a later stage.

SUSE Here is a short example of a autoinst.xml file which adds a new user during setup.

```
<profile xmlns="http://www.suse.com/1.0/yast2ns"
         xmlns:config="http://www.suse.com/1.0/configns">
 <users config:type="list">
   <user>
     <username>jdoe</username>
     <user_password>t0psEcreT</user_password>
     <encrypted config:type="boolean">false</encrypted>
     <forename>John</forename>
     <surname>Doe</surname>
   </user>
 </users>
</profile>
```

Do not omit the xmlns and xmlns:config attributes of the profile tag.

Troubleshooting:

If the OS configurations in the deployed operating system are not what you expected, you must examine the parameter files. They are the result of the merge between the custom file and the default file created.

Red Hat To troubleshoot OS configuration parameters after a failed deployment, complete the following procedure:

- Without rebooting the target:
  1. Type Alt+F2 on the target. This opens a shell.
  2. In the opened shell, view the file /tmp/anaconda.log.
- Look for ks.cfg at the root of the partition labeled rembo. The file contains the information merged from the custom and the default files.

SUSE To troubleshoot OS configuration parameters after a failed deployment, there are the following options:

- Without rebooting the target:
  1. Type Alt+F2 on the target. This opens a shell.
  2. In the opened shell, view the file/var/log/YaST2/y2log .
- You must look for autoinst.xml at the root of the partition labelled rembo. The file contains the information merged from the custom and the default files.

Software modules for Linux operating systems on PowerPC

Software modules are images other than system profiles that can be created to address various needs.

Tivoli Provisioning Manager for Images is based on imaging technology. As administrator, you create images of components that you want to see on every target, and the automated deployment merges and restores these images on each target, automatically, when needed.

Tivoli Provisioning Manager for Images can handle most scenarios for software deployment and post-installation configuration.

Types of software modules

There are many types of software modules. Depending on the type of package and installation files, the wizard guides you through the different steps to achieve your software module with minimal effort. The types of software package supported by the wizard are listed in this section.

- **A Linux application installation, using RPM**
- **A custom action on the target computer**. This includes OS configuration changes such as commands to be run, and copying sets of files on the target.

Creating software modules

There are distinct types of software modules which vary according to the operating system being deployed. The software wizard guides you through the creation of software modules for each type.

Creating software modules with RPM for Linux operating systems:

Using RPM for Linux software installations.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **Linux** and click **Next**.
4. Select **A Linux application installation, using RPM** and click **Next**.
5. Complete the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

     For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=xh`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

```
install /sPB /rs /rps /l <=1h
```

  - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a custom action software module for Linux operating systems:

Software modules can also contain custom actions to be performed on the target.

They are divided into:
- An OS configuration change to perform on the target
- A set of files to copy on the target

Configuration changes are further subdivided. Depending on the operating system, you can:
- Copy a single text file
- Run a single command file, this can be a batch file or a vb script file.
- Boot a virtual floppy disk

In the OS configuration change wizard screen, you can select **Activate keyword substitutions**. If you use this option, you can specify which keywords must be substituted in the software module details.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select the operating system and click **Next**.
4. Select **A custom action on the target** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when your software module must be installed: when the OS is installed, or after one or more additional reboot. Most of the time, you must install the software module at the same time as the operating system. However, you can decide to install them in a specified order to avoid software-specific conflicts.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The full path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard automatically suggests the appropriate command line to run the installation unattended. However, you might need to add some additional parameters to the command.

For example, you can specify an hour parameter to cancel an activity, if the activity does not complete before the end of the specified time. The parameter syntax format is `<=xh`, where `x` is an integer representing the number of hours after which the activity is canceled. In the following example you can specify to cancel an application installation if the installation process has not completed after one hour, by adding `<=1h` at the end of the command line:

```
install /sPB /rs /rps /l <=1h
```

- The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

Repeating custom actions:

Some commands must be run every time the target boots during a deployment.

This is typically the case if you want to repeatedly connect a network share. This connection is destroyed when rebooting. You can therefore create a single software module with a `netuse` command to set the network share and set this software module to run once after each reboot, starting at a specific reboot.

This option is available for executing a single command.

1. Create your software module.
2. Double-click on the software module name in the **Software components** page to obtain the **Software details** page
3. Click **Edit** in the title of the **Package information** section.
4. Select the installation stage at which the software module must be applied first.
5. Select **Run at each software pass until end of deployment** and click **OK**.

Creating a software group:

Simplify the management of your software modules by grouping them into containers called *software groups*.

A *software group* is a collection of software modules that behaves as a standard software module.

The advantage of software groups is to manipulate only one object instead of several software modules when they should all behave in the same way. For example, you can select a whole software group for deployment, create a binding rule for it, or change its software application order, instead of doing it for each software module individually.

The elements of a software group are individual software modules. You cannot nest software groups within software groups.

A software module can belong to several software groups simultaneously.

To create a software group:

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.

3. Select **A software group** and click **Next**.
4. Select all the software modules that you want to include in your software group and click **Next**.
5. Follow the remaining instructions of the wizard to create your software group.

You can now create binding rules for your software group, modify its application order, export it to a RAD file, or use it in a deployment, as if it were a standard software module.

You can also edit the software group, for example to add or remove software modules.

Editing software modules

You can edit the basic parameters of a software module, upload new files into your software module, and update drivers.

1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
2. From **Software details** page, use the links and buttons. Links vary according to the type of software module. Not all the links listed are necessary available.
   - To edit the base parameters of a software module, click **Edit** at the top of the **Software module information** section.
   - To update files or add new files into the software module, click **Edit software module files**, or a link with a similar name, and select **Upload file** from the contextual menu.

     Note: File upload is limited to 16 MB.
   - For software groups, to add or remove software modules:
     a. Click **Edit** at the top of the **Software group contents** section.
     b. Select the software modules that you want to add.
     c. Deselect the software modules that you want to remove.
     d. Click **OK**.

Keeping command lines confidential

When you use command lines in your software modules, their call and their output are stored in deployment logs. In some circumstances, for example when the command line includes a password or a product key, it might be necessary to keep the information contained in the command line confidential. Three levels of confidentiality are available.

No confidentiality

The command line is visible in the web interface and on the target during the installation, its call is logged, and its output is also logged.

The command line call is not logged

The command line is visible in the web interface, and its output is logged, but the command line call, containing the whole command line string with all parameters, is visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by one exclamation mark (`!`).

The command line call and output are not logged

The command line is visible in the web interface, but its call and output are visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by two exclamation marks (`!!`).

To keep command lines confidential:

- Enter the appropriate number of exclamation points in front of the command in the Software Wizard when first creating the software module.
- Edit the software module information
  1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
  2. Click **Edit** in the Software module information banner.
  3. Update the command line with the appropriate number of exclamation points.
  4. Click **OK**.

Keyword substitution

You can usefully use keyword which act as variables and are substituted with their values during deployments. Keywords can either refer database values or server specific values, given by the user.

Syntax

Variable substitution expressions follow the syntax given here. They start with the character { and end on the same line with }. Words between these two characters are interpreted by using one of the following schemes:

- *{$expr$}* the expression is replaced with the string resulting of the evaluation of `expr`.
- *{/expr/ab}* the expression is replaced with the string resulting of the evaluation of `expr`, but each occurrence of the character `"a"` is replaced by the character `"b"` (character-based substitution).
- *{=expr=test content=this is a test}* the text `"this is a test"` is included in the destination file only if the string resulting of the evaluation of `expr` is equal to the text `"test content"`.
- *{!expr!test content!this is a test}* the text `"this is a test"` is included in the destination file only if the string resulting of the evaluation of `expr` is not equal to the text `"test content"`.

Note: If a variable does not exist (for example, it contains a typing error or it is not described in `server.ini`) but it is used in a command, its value is supposed to be empty which can result in deployment errors.

Database keywords

Within an expression, database records can be referred to. Within a record, each field can be accessed using the standard C notation (`record.fieldname`). The exhaustive list of these fields can be obtained from the database records, with the following correspondences between variable and database record names:

Table 13. Records for `free-text conditions`

Variable record name	Database record name
Disk	DiskInventory
DMI	DMIInventory
Order	BOM

Table 13. Records for `free-text conditions` *(continued)*

Variable record name	Database record name
User	UserProfile
System	SystemProfile
PCI	PCIInventory

Below are a few examples of available fields:

- `Order.IP`: a string, the target IP address, such as `192.168.1.2`
- `Order.MAC`: a string, the target MAC address, such as `00:01:02:03:04:05`
- `Order.SN`: a string, the target Serial Number, such as `CH12345678`
- `Order.Model`: a string, the computer model name, such as `e-Vectra`
- `User.UserCateg0`: a string, without any restriction, such as`technicians`
- `DMI.Vendor`: a string, the vendor name, such as `Hewlett-Packard`
- `DMI.Product`: a string, same as `Order.Model`
- `DMI.ProcModel`: a string, the processor model
- `Disk[0].Type`: a string, the disk 0 drive type, such as `ATAPI`
- `Disk[0].Media`: a string, the disk 0 media type, such as `Disk` or `CD`
- `Disk[0].DiskSize`: a number, the physical size of the disk (if detected)
- `PCI[0].VendorID`: a string, the hexadecimal vendor ID of the device
- `PCI[0].DeviceID`: a string, the hexadecimal device ID of the device

For disks and PCI devices, you can use the function `sizeof` (`sizeof(Disk)` and `sizeof(PCI)`) to discover the number of devices present. You can then use indexes to access these devices.

As an example for keyword substitution, if `BomID` has `OrgName Rembo SaRL`, `RemboServer 192.168.168.16`, and `IP 192.168.168.32` for value `1`, the following text

```
BomID:{$Order.BomID$}
OrgName:{$User.OrgName$}/{$StrToLower(User.OrgName)$}
RemboServer:{$Order.RemboServer$}
IP:{$Order.IP$}
```

gives the following results after keywords are substituted (note the use of a Rembo-C function within the expression to be substituted):

```
BomID:1
OrgName:Rembo SaRl/rembo sarl
RemboServer:192.168.168.16
IP:192.168.168.32
```

Server specific keywords

If you want to set up server specific keywords, which are defined exclusively by the user and per server, you must edit `Files/global/rad/server.ini`.

Start the file with `[Custom]` and add a line per keyword, in the format **keyword=value**, where `keyword` is a word of your choice and value the `value` you want to give it.

To use the keyword in a command, type `Server.keyword` and activate keyword substitution when creating the software module.

Note: `server.ini` is not replicated between servers. If you use multiple servers, you must edit `server.ini` on each server.

Customizing the software page

You can view the software modules in a tree viewer or in a list viewer. The list viewer allows you to customize the visible information.

You must have created at least one software module, otherwise there is nothing to view.

To customize the visible information

1. Go to **Server** > **OS deployment** > **Software modules**. Then click **List view**.
2. From the list view, you can
   - Drag the column separator in the column heading to resize the column.
   - Click on the triangular arrow to the left of the column name to sort the software modules by column criteria.
   - Click on the arrow on the right of the column name and select an option to filter the information. Filtering on several columns is cumulative.
3. For more options, right click anywhere to open the contextual menu and select **Arrange columns**.
   - Select the columns you want to see and clear the others.
   - Click on the minus or plus icons to decrease or increase the size of a column.
   - Select a column and use the up and down arrows to move the column relatively to the others.

   Click **OK** to save your changes. The updated version of the list view is visible in the **Software modules** page.

To return to the tree view, click **Tree view**. You can also access the details of the software modules by double-clicking on a software module name, from either view.

OS configuration and software bindings

OS configuration bindings determine which configurations are available to a target when booting the target on the network, while software bindings correspond to the list of software modules currently assigned to the target.

OS configuration and software bindings are created when:

- The Target Monitor has been used to manually modify OS configuration and software bindings for the target
- A deployment has been started with the Target Monitor. In this case, an OS configuration binding is added for the corresponding OS configuration.
- Automatic binding rules are configured in the **Details** page of OS configurations or software modules. Some of these rules have matching values for the specified criteria. These bindings cannot be modified, except by modifying the rules.

With the Target Monitor, you can browse, remove or add OS configuration and software bindings to any target present in the database. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Binding software modules and OS configurations to targets:

Bindings link software modules and OS configurations to targets to enable automatic deployment. When binding to targets, you explicitly provide the list of software modules and OS configurations to bind to your target.

To explicitly bind a software module or a OS configurations to a target, there are two methods:

- From the **Target Monitor** page
- From the **Target details** page

If you want to bind software modules or OS configurations to a group of targets, you must do it through the Target Monitor.

From the Target Monitor:

1. Select a target or a group of targets
2. Select **Bind software** or **Bind OS configurations** from the contextual menu
3. Select the items to bind from the popup window
4. Click **OK**

From the Target details page:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.
2. Go to the **Bindings** panel.
3. Click **Edit** in the relevant section to add explicit bindings for OS configurations and software modules.
4. Select the items for which you want to add explicit bindings.
5. Click **OK**

You can also clear items to remove their explicit bindings. To remove a binding by rule, you must modify the rule.

Binding software modules to a deployment scheme:

Software modules can be bound to deployment schemes.

Take a company with offices in three locations: New York, Quebec City, and Mexico City. In each of these locations, the company has people in human resources, sales, logistics, and product development. For the sake of simplicity, consider further that all the employees use either one of two types of computers: a desktop, or a notebook. All desktop computers are identical (with the same network card, system board, disks, and so on) and the same applies for all notebooks.

In this scenario, the company needs two profiles, one with the image for notebooks and one with the image for desktop computers. Three configurations per profile (six in total) are necessary to integrate the different parameters of the different locations, in particular language and time zone information. Finally, schemes are set according to the employees' department, with software modules specific to the different departments bound directly to the deployment schemes.

1. Go to **Server** > **OS deployment** > **Task templates** Select the **Deployment Schemes** folder. Double-click on a deployment scheme to view its details.

2. Click **Edit** on the **Software bindings** section of the page to open the dialog to bind software modules to schemes.
3. Select which software modules you want to bind to your deployment scheme, in addition to software modules that can have been bound to targets.
4. *(Optional)* If you want to use only the software checked in the window when deploying with this scheme, select the **Discard all other software binding rules** check box.

Automatic binding rules:

Automatic binding rules are used to create bindings between OS configurations and targets, or software modules and targets, without having to specifically bind a OS configuration or a software module on each target.

Rules are created in OS configurations and software modules to determine which targets are automatically bound to the OS configuration or software module.

Rules are made of criteria and values. If a target has a matching value for all criteria in the rule, the OS configuration or software module will be bound to that target. The binding will be displayed with the mention **by rule** in the OS configuration panel of the target properties for targets that match the criteria. For example, if the criteria is the model name, and the value is `Optiplex`, targets with a model name starting with `Optiplex` will be bound to the object where the rule has been defined.

Automatic binding rules are defined in Tivoli Provisioning Manager for OS deployment at the bottom of the **OS configuration details** or **Software details** page.

To create a new binding rule, click **New rule** located at the bottom of the Web interface:

1. The dialog displayed to create a new binding rule is different depending on whether you are adding a rule to an OS configuration or to a software module. When adding a binding rule to a software module, you can set values for the following criteria:
   - A deployment scheme
   - A system profile
   - A current OS configuration
   - Administrative group
   - One of the system-definable and user-definable fields of the database (only used if you have customized the database)
   - An operating system type, such as Windows 2000
   - An operating system version, such as SP2
   - An operating system language
   - An operating system architecture, such as x86-32
   - A computer model name
   - A BIOS version
   - A PCI device
   - A base board
   - MultiChassi
   - HAL Type

- A free-text condition in Rembo-C; syntax

  For example, to create a binding based on the operating system type between a software module and targets, you must create a new rule, click **OS type**, and select the operating system version that you want to limit this software module to.

2. When adding a binding rule to an OS configuration, you can set a condition on the deployment scheme, and on the computer model name. The next ten fields are only used if you have customized your database and want to match specific user categories.
3. Finally, you can enter a free-text condition following the Rembo-C; syntax. They must only be used by advanced users.

   The conditions determine the applicability of the rule and evaluate to `true` or `false`. A condition must be formed using the variables also used for keyword substitutions in software modules, combined with Java-like logical operators, listed by order of priority in the table:

Table 14. Logical operators for free-text conditions

Operator	Meaning
<	smaller than
<=	smaller than or equal to
=>	greater than or equal to
>	greater than
==	equal to
!=	not equal to
&&	AND operator
||	OR operator

For example, a typical condition can be:

```
Disk[0].DiskSize > 10*1024*1024
```

Note: If a condition cannot be evaluated, it is considered to have the value `false`.

Scheduling the application of software modules

It is not possible to schedule the application of software modules for this operating system and hardware.

All software modules are applied **When the OS is installed**, regardless of the set stage in the **Software application order** window. Reboots are not handled either.

Task templates for Linux operating systems on PowerPC

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout
The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout
The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout
The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout
The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

Deployment schemes are not linked to architecture of the target or the operating system. Therefore, the deployment scheme wizard always offers to set all modifiable parameters. When deploying, parameters incompatible with either the architecture of the target or the operating system being deployed are not taken into account.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard, by clicking **New deployment scheme** from the **Task templates** page.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   The following parameters apply for simple one-time deployments and for redeployment operations.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, therefore the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters
   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

   Edit parameters for targets that are not yet in the database
   to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

   Never edit parameters
   to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

   You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

Yes, display a warning
if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment
if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, becauseTivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

Note: These parameters have no effect on PowerPC and Cell targets.

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of target computers to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed computer, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, /. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

Note: Multicast is available only if

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of computers that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast

to deploytargets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast, without explicit synchronization

to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.

Multicast, with synchronization on all targets
to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Multicast, with the following synchronization parameters
to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

You can use a network share or Linux HTTP protocol on the server to download the files to the target computers, rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP protocol provides a shorter operating system installation time. To use a network share or Linux HTTP protocol:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module). .

On-site deployment

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment parameters

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
Indicate if you want to keep the deployment image in a protected partition and the size of this partition.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation

This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user.

Unbind software module at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is currently relevant forWindows Vista/2008/7/2012deployments only.

Send mail at end

This option can be set only if a *sendmail* tunnel has been created.

Send mail to

This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network

Note: This parameter has no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button

This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Deploying Linux on PowerPC

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

When you deploy a cloning system profile, you may encounter driver and/or disk mapping issues if the hardware of the reference target of the cloning system profile is significantly different from the hardware of the deployment target. This is particularly true for Linux operating systems where driver binding is not available.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. Select **Simple deployment** and click **Next**
5. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Note: The Wake on LAN options are available only on Intel targets.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor:

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for Linux on PowerPC:

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification:

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     – **No**
     – **Whatever the notification is**
     – **If deployment is successfully completed**
     – **If deployment failed**
  5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.

- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Provisioning Solaris on SPARC targets

Deploying Solaris operating systems has a number of specificities and prerequisites.

Note: Tivoli Provisioning Manager for Images does not support the operating system deployment on Fujitsu SPARC targets.

To deploy Solaris, you must have installed a Solaris install server which is also running the web interface extension.

1. Set up and configure a Solaris install server.
   a. Set up a Solaris install server
   b. Configure it for operating system content.
   c. Configure it for Flash Archive content.
   d. Install the web interface extension on the Solaris install server.
2. Register new targets. You must add SPARC targets manually into the Target Monitor or import a comma-separated text file containing a list of targets to be added.
3. Setup the specific SPARC DHCP options for these targets.

You can then create your Solaris system profiles and software modules.

You can then also boot SPARC targets on the OS deployment server.

System profiles for Solaris operating systems

A system profile is the partition layout and list of files to deploy an operating system, either by unattended setup or by cloning, from a reference target or from a reference image file.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- Disk cloning is not supported for Linux PowerPC and Cell targets. Only unattended setup is supported.
- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.
- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.
- You cannot deploy Linux profile with an LVM root partition if you use deployment media.

Creating an unattended setup system profile for Solaris operating system

You can install operating systems using standard installation processes in unattended mode.

- You must have set up a Solaris install server, as described in the Installation Guide, Chapter 6, sectioninstall/tosd_solariscontent.dita.
- If you want to create a system profile from Solaris 10 Update 6 or higher, do not forget to modify the wanboot directory.
  - Create a directory named `interim_dir` by running:
    ```
    mkdir /export/install/sol10-miniroot/interim_dir
    ```
  - Copy the `platform` subdirectory from `Solaris_10/Tools/Boot` into the `/sol10-miniroot/interim_dir` directory as follows:
    ```
    (cd /export/install/Solaris_10/Tools/Boot ; tar cf - platform) |
    (cd /export/install/sol10-miniroot/interim_dir ; tar xvf - )
    ```

- You must create your system profile from a Solaris target and the NFS server must also reside on a Solaris target.
- Make sure the web interface extension is running.

1. Go to **Server** > **OS deployment** > **System profiles**.
2. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
3. Select **Unattended setup** in the first pane of the profile wizard.
4. Select **A Solaris system profile**.
5. Follow the instruction of the profile wizard.

When your first unattended installation profile is created, you can use it to deploy targets.

Now that you have created your Solaris unattended profile, you can optionally move your Solaris install server to a UNIX target with an NFS server compatible with Solaris targets. In this case, you must edit your profile configuration to update the value of **NFS install source** to the new NFS server.

Note: When using a Linux NFS server, the NFS share should force to NFS 3 since NFS 4 from Solaris is not compatible with NFS 4 from Linux.

Creating a system profile from a Solaris Flash archive

You can create a cloning system profile from a Solaris Flash archive (a file with a `.flar` extension).

To be able to create your system profile, you need not only the Solaris Flash archive on your NFS server, but also the complete installation files for a Solaris operating system. The Profile Wizard asks you first for the directory in which the operating system installation files are located. It checks whether the `.cdtoc` hidden file is present before asking you for the exact location of the Flash Archive you want to use for your system profile creation.

To create a system profile from a reference image, you must follow these steps:

1. Go to **Server** > **OS deployment** > **System Profiles**.
2. Click **New Profile**. This opens a system profile wizard that guides you through the steps of creating a profile.
3. Select **Cloning from a reference image file** and click **Next**.
4. Select the corresponding image format and click **Next**.
5. Follow the instruction of the profile wizard.

Creating Flash archives:

Although Tivoli Provisioning Manager for Images is not involved in the creation of Flash archives, the process is described for convenience.

For more information, see the SUN Solaris documentation.

Creating flash archives in Solaris is a relatively simple process.

1. Mount the flash archive directory on the install server.
   a. Create a local mount point, a directory that you can reference locally.
      ```
      mkdir /export/flash
      ```
   b. Mount the remote flash archive directory

```
mount certdev-sun2:/export/flars /export/flash
```

2. Run the flash archive creation command

   ```
   flarcreate -n flarname.flar -x /export/flash -c /export/flash/flarname.flar
   ```
3. Restart the computer to make sure that all unnecessary file handles are closed.
4. Check that the new flash archive is created and sent to the Flash directory of the Solaris install Server.

Note: There can be installation specific issues with Flash archives. In particular, some symbolic links may prevent flash archives to be restored properly. As a workaround, remove the symbolic links and copy the actual files in the appropriate directory.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles:
To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.
3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/c`*x*`d`*x*, where *x* is the disk number. Edit your device mappings accordingly.

4. Click **OK**.
5. Repeat step 2 on page 262 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies unique.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Editing OS configuration parameters in the web interface for Solaris:

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:

1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Editing a Solaris parameter file:

You can modify OS configuration parameters by editing a file. This option allows you to modify parameters that are not displayed in the web interface. However, you must be experienced to use this option advantageously, because Tivoli Provisioning Manager for Images does not provide any syntax checking of the file. Information about the file format and syntax can be found in the documentation of the operating system itself.

1. Click **Edit custom 'solaris.profile'** to edit the file.
2. Type the parameters and their values in the syntax requested by the operating system, or copy and paste it from another editor.
3. Click **OK**.

Tivoli Provisioning Manager for Images merges the information of the edited file with the information provided on the web interface (default file). Unless otherwise specified, parameters specified in the default file override the content of the custom file.

Here is an example of a disk layout described in a `solaris.profile` file:

```
partitioning    explicit
filesys         rootdisk.s0     free    /
filesys         rootdisk.s1     2048    swap
cluster         SUNWCpm         delete
cluster         SUNWCpmx        delete
cluster         SUNWCdial       delete
cluster         SUNWCdialx      delete
cluster         SUNWCadm
cluster         SUNWCcpc
```

By default the deployment provides its own pre-installation and post-installation scripts for generating profiles dynamically and installing software modules specified in the database.

If you want to add your own code in the pre-installation and post-installation scripts, you can do so by adding sections in the custom profile configuration file `solaris.profile`.

```
SI_BEGIN:
echo 'This is the pre-installation script'
...
SI_PROFILE:
echo 'This is the profile configuration'
partitioning    explicit
filesys         rootdisk.s0     free    /
filesys         rootdisk.s1     2048    swap
cluster         SUNWCpm         delete
cluster         SUNWCpmx        delete
cluster         SUNWCdial       delete
cluster         SUNWCdialx      delete
cluster         SUNWCadm
cluster         SUNWCcpc
SI_FINISH:
echo 'This is the post-installation script'
...
```

Note: The **`cluster`** command is not supported in `solaris.profile` files attached to cloning system profiles.

Software modules for Solaris operating systems

Software modules are images other than system profiles that can be created to address various needs.

Tivoli Provisioning Manager for Images is based on imaging technology. As administrator, you create images of components that you want to see on every target, and the automated deployment merges and restores these images on each target, automatically, when needed.

Tivoli Provisioning Manager for Images can handle most scenarios for software deployment and post-installation configuration.

Types of software modules

There are many types of software modules. Depending on the type of package and installation files, the wizard guides you through the different steps to achieve your software module with minimal effort. The types of software package supported by the wizard are listed in this section.

- **A Solaris package installation, using pkgadd**
- **A custom action on the target computer**. This includes OS configuration changes such as commands to be run, and copying sets of files on the target.

Creating software modules

There are distinct types of software modules which vary according to the operating system being deployed. The software wizard guides you through the creation of software modules for each type.

Creating Solaris software modules with pkgadd:

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select **Solaris** and click **Next**.
4. Select **A Solaris package installation, using pkgadd** and click **Next**.
5. Follow the instructions of the wizard to create your software module

   Note: Make sure the folder containing Solaris package also includes the corresponding `pkginfo`. The software module cannot be created if `pkginfo` is not found in the folder.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:

   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when yoursoftware module must be installed. Solaris software modules must always be installed with the operating system.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard suggests automatically the appropriate command line to run the installation unattended. However, you might must add some additional parameters to the command.

     All software packages executed during installation run in a specific environment where the newly installed system is mounted under directory `/a`. To write a file in the root directory, you must use the `/a` path. The `/a` prefix is automatically added to the destination path when copying packages, so this only applies to command lines referring to specific paths on the newly installed system
   - The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the

deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

You can organize your software modules by creating software module subfolders following the same procedure as for system profiles.

Creating a custom action software module for Solaris operating systems:

Software modules can also contain custom actions to be performed on the target.

They are divided into:
- An OS configuration change to perform on the target
- A set of files to copy on the target

Configuration changes are further subdivided. Depending on the operating system, you can:
- Copy a single text file
- Run a single command file.

In the OS configuration change wizard screen, you can select **Activate keyword substitutions**. If you use this option, you can specify which keywords must be substituted in the software module details.

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software** to run the software wizard.
3. Select the operating system and click **Next**.
4. Select **A custom action on the target** and click **Next**.
5. Follow the instructions of the wizard to create your software module.

   Parameters of the software module are pre-filled for you but they can be modified in the appropriate step of the software wizard. These parameters include:
   - A description that identifies the software module in the software module tree.
   - A comment with additional information about the software module.
   - The stage of the deployment when yoursoftware module must be installed. Solaris software modules must always be installed with the operating system.
   - A file name to store your image on the OS deployment server. Software modules typically have a `.pkg` extension.
   - The full path to where the installation files are restored on the target. This path is relative to the system root partition.
   - An additional command line that might be necessary to install your software module. When possible, the wizard suggests automatically the appropriate command line to run the installation unattended. However, you might must add some additional parameters to the command.

     All software packages executed during installation run in a specific environment where the newly installed system is mounted under directory `/a`. To write a file in the root directory, you must use the `/a` path. The `/a` prefix is automatically added to the destination path when copying packages, so this only applies to command lines referring to specific paths on the newly installed system

- The operating system with which the software module is compatible. The deployment wizard offers to deploy only software modules compatible with the operating system being deployed. Moreover, this parameter prevents the deployment of a bound software module if the it is not compatible with the operating system. Additionally, you can also sort and filter software modules by this parameter in list view.

Repeating custom actions:

Some commands must be run every time the target boots during a deployment.

This is typically the case if you want to repeatedly connect a network share. This connection is destroyed when rebooting. You can therefore create a single software module with a `netuse` command to set the network share and set this software module to run once after each reboot, starting at a specific reboot.

This option is available for executing a single command.

1. Create your software module.
2. Double-click on the software module name in the **Software components** page to obtain the **Software details** page
3. Click **Edit** in the title of the **Package information** section.
4. Select the installation stage at which the software module must be applied first.
5. Select **Run at each software pass until end of deployment** and click **OK**.

Creating a software group:

Simplify the management of your software modules by grouping them into containers called *software groups*.

A *software group* is a collection of software modules that behaves as a standard software module.

The advantage of software groups is to manipulate only one object instead of several software modules when they should all behave in the same way. For example, you can select a whole software group for deployment, create a binding rule for it, or change its software application order, instead of doing it for each software module individually.

The elements of a software group are individual software modules. You cannot nest software groups within software groups.

A software module can belong to several software groups simultaneously.

To create a software group:

1. Go to **Server** > **OS deployment** > **Software modules**.
2. Click **New software**.
3. Select **A software group** and click **Next**.
4. Select all the software modules that you want to include in your software group and click **Next**.
5. Follow the remaining instructions of the wizard to create your software group.

You can now create binding rules for your software group, modify its application order, export it to a RAD file, or use it in a deployment, as if it were a standard software module.

You can also edit the software group, for example to add or remove software modules.

Editing software modules

You can edit the basic parameters of a software module, upload new files into your software module, and update drivers.

1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.
2. From **Software details** page, use the links and buttons. Links vary according to the type of software module. Not all the links listed are necessary available.
   - To edit the base parameters of a software module, click **Edit** at the top of the **Software module information** section.
   - To update files or add new files into the software module, click **Edit software module files**, or a link with a similar name, and select **Upload file** from the contextual menu.

     Note: File upload is limited to 16 MB.
   - For software groups, to add or remove software modules:
     a. Click **Edit** at the top of the **Software group contents** section.
     b. Select the software modules that you want to add.
     c. Deselect the software modules that you want to remove.
     d. Click **OK**.

Keeping command lines confidential

When you use command lines in your software modules, their call and their output are stored in deployment logs. In some circumstances, for example when the command line includes a password or a product key, it might be necessary to keep the information contained in the command line confidential. Three levels of confidentiality are available.

No confidentiality
The command line is visible in the web interface and on the target during the installation, its call is logged, and its output is also logged.

The command line call is not logged
The command line is visible in the web interface, and its output is logged, but the command line call, containing the whole command line string with all parameters, is visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by one exclamation mark (`!`).

The command line call and output are not logged
The command line is visible in the web interface, but its call and output are visible in the logs neither on the web interface nor on the target.

To apply this level of confidentiality, you must prefix the command line by two exclamation marks (`!!`).

To keep command lines confidential:

- Enter the appropriate number of exclamation points in front of the command in the Software Wizard when first creating the software module.
- Edit the software module information
  1. Go to **Server** > **OS deployment** > **Software modules**. Double-click on a software module to view the details.

2. Click **Edit** in the Software module information banner.
3. Update the command line with the appropriate number of exclamation points.
4. Click **OK**.

Keyword substitution

You can usefully use keyword which act as variables and are substituted with their values during deployments. Keywords can either refer database values or server specific values, given by the user.

Syntax

Variable substitution expressions follow the syntax given here. They start with the character { and end on the same line with }. Words between these two characters are interpreted by using one of the following schemes:

- *{$expr$}* the expression is replaced with the string resulting of the evaluation of `expr`.
- *{/expr/ab}* the expression is replaced with the string resulting of the evaluation of `expr`, but each occurrence of the character "`a`" is replaced by the character "`b`" (character-based substitution).
- *{=expr=test content=this is a test}* the text "`this is a test`" is included in the destination file only if the string resulting of the evaluation of `expr` is equal to the text "`test content`".
- *{!expr!test content!this is a test}* the text "`this is a test`" is included in the destination file only if the string resulting of the evaluation of `expr` is not equal to the text "`test content`".

Note: If a variable does not exist (for example, it contains a typing error or it is not described in `server.ini`) but it is used in a command, its value is supposed to be empty which can result in deployment errors.

Database keywords

Within an expression, database records can be referred to. Within a record, each field can be accessed using the standard C notation (`record.fieldname`). The exhaustive list of these fields can be obtained from the database records, with the following correspondences between variable and database record names:

Table 15. Records for `free-text conditions`

Variable record name	Database record name
Disk	DiskInventory
DMI	DMIInventory
Order	BOM
User	UserProfile
System	SystemProfile
PCI	PCIInventory

Below are a few examples of available fields:

- `Order.IP`: a string, the target IP address, such as `192.168.1.2`
- `Order.MAC`: a string, the target MAC address, such as `00:01:02:03:04:05`
- `Order.SN`: a string, the target Serial Number, such as `CH12345678`

- Order.Model: a string, the computer model name, such as e-Vectra
- User.UserCateg0: a string, without any restriction, such astechnicians
- DMI.Vendor: a string, the vendor name, such as Hewlett-Packard
- DMI.Product: a string, same as Order.Model
- DMI.ProcModel: a string, the processor model
- Disk[0].Type: a string, the disk 0 drive type, such as ATAPI
- Disk[0].Media: a string, the disk 0 media type, such as Disk or CD
- Disk[0].DiskSize: a number, the physical size of the disk (if detected)
- PCI[0].VendorID: a string, the hexadecimal vendor ID of the device
- PCI[0].DeviceID: a string, the hexadecimal device ID of the device

For disks and PCI devices, you can use the function sizeof (sizeof(Disk) and sizeof(PCI)) to discover the number of devices present. You can then use indexes to access these devices.

As an example for keyword substitution, if BomID has OrgName Rembo SaRL, RemboServer 192.168.168.16, and IP 192.168.168.32 for value 1, the following text

```
BomID:{$Order.BomID$}
OrgName:{$User.OrgName$}/{$StrToLower(User.OrgName)$}
RemboServer:{$Order.RemboServer$}
IP:{$Order.IP$}
```

gives the following results after keywords are substituted (note the use of a Rembo-C function within the expression to be substituted):

```
BomID:1
OrgName:Rembo SaRl/rembo sarl
RemboServer:192.168.168.16
IP:192.168.168.32
```

Server specific keywords

If you want to set up server specific keywords, which are defined exclusively by the user and per server, you must edit Files/global/rad/server.ini.

Start the file with [Custom] and add a line per keyword, in the format **keyword=value**, where keyword is a word of your choice and value the value you want to give it.

To use the keyword in a command, type Server.keyword and activate keyword substitution when creating the software module.

Note: server.ini is not replicated between servers. If you use multiple servers, you must edit server.ini on each server.

Customizing the software page

You can view the software modules in a tree viewer or in a list viewer. The list viewer allows you to customize the visible information.

You must have created at least one software module, otherwise there is nothing to view.

To customize the visible information

1. Go to **Server** > **OS deployment** > **Software modules**. Then click **List view**.

2. From the list view, you can
   - Drag the column separator in the column heading to resize the column.
   - Click on the triangular arrow to the left of the column name to sort the software modules by column criteria.
   - Click on the arrow on the right of the column name and select an option to filter the information. Filtering on several columns is cumulative.
3. For more options, right click anywhere to open the contextual menu and select **Arrange columns**.
   - Select the columns you want to see and clear the others.
   - Click on the minus or plus icons to decrease or increase the size of a column.
   - Select a column and use the up and down arrows to move the column relatively to the others.

   Click **OK** to save your changes. The updated version of the list view is visible in the **Software modules** page.

To return to the tree view, click **Tree view**. You can also access the details of the software modules by double-clicking on a software module name, from either view.

OS configuration and software bindings

OS configuration bindings determine which configurations are available to a target when booting the target on the network, while software bindings correspond to the list of software modules currently assigned to the target.

OS configuration and software bindings are created when:
- The Target Monitor has been used to manually modify OS configuration and software bindings for the target
- A deployment has been started with the Target Monitor. In this case, an OS configuration binding is added for the corresponding OS configuration.
- Automatic binding rules are configured in the **Details** page of OS configurations or software modules. Some of these rules have matching values for the specified criteria. These bindings cannot be modified, except by modifying the rules.

With the Target Monitor, you can browse, remove or add OS configuration and software bindings to any target present in the database. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.

Binding software modules and OS configurations to targets:

Bindings link software modules and OS configurations to targets to enable automatic deployment. When binding to targets, you explicitly provide the list of software modules and OS configurations to bind to your target.

To explicitly bind a software module or a OS configurations to a target, there are two methods:
- From the **Target Monitor** page
- From the **Target details** page

If you want to bind software modules or OS configurations to a group of targets, you must do it through the Target Monitor.

From the Target Monitor:

1. Select a target or a group of targets

2. Select **Bind software** or **Bind OS configurations** from the contextual menu
3. Select the items to bind from the popup window
4. Click **OK**

From the Target details page:

1. Go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details.
2. Go to the **Bindings** panel.
3. Click **Edit** in the relevant section to add explicit bindings for OS configurations and software modules.
4. Select the items for which you want to add explicit bindings.
5. Click **OK**

You can also clear items to remove their explicit bindings. To remove a binding by rule, you must modify the rule.

Binding software modules to a deployment scheme:

Software modules can be bound to deployment schemes.

Take a company with offices in three locations: New York, Quebec City, and Mexico City. In each of these locations, the company has people in human resources, sales, logistics, and product development. For the sake of simplicity, consider further that all the employees use either one of two types of computers: a desktop, or a notebook. All desktop computers are identical (with the same network card, system board, disks, and so on) and the same applies for all notebooks.

In this scenario, the company needs two profiles, one with the image for notebooks and one with the image for desktop computers. Three configurations per profile (six in total) are necessary to integrate the different parameters of the different locations, in particular language and time zone information. Finally, schemes are set according to the employees' department, with software modules specific to the different departments bound directly to the deployment schemes.

1. Go to **Server** > **OS deployment** > **Task templates** Select the **Deployment Schemes** folder. Double-click on a deployment scheme to view its details.
2. Click **Edit** on the **Software bindings** section of the page to open the dialog to bind software modules to schemes.
3. Select which software modules you want to bind to your deployment scheme, in addition to software modules that can have been bound to targets.
4. *(Optional)* If you want to use only the software checked in the window when deploying with this scheme, select the **Discard all other software binding rules** check box.

Automatic binding rules:

Automatic binding rules are used to create bindings between OS configurations and targets, or software modules and targets, without having to specifically bind a OS configuration or a software module on each target.

Rules are created in OS configurations and software modules to determine which targets are automatically bound to the OS configuration or software module.

Rules are made of criteria and values. If a target has a matching value for all criteria in the rule, the OS configuration or software module will be bound to that target. The binding will be displayed with the mention **by rule** in the OS configuration panel of the target properties for targets that match the criteria. For example, if the criteria is the model name, and the value is `Optiplex`, targets with a model name starting with `Optiplex` will be bound to the object where the rule has been defined.

Automatic binding rules are defined in Tivoli Provisioning Manager for OS deployment at the bottom of the **OS configuration details** or **Software details** page.

To create a new binding rule, click **New rule** located at the bottom of the Web interface:

1. The dialog displayed to create a new binding rule is different depending on whether you are adding a rule to an OS configuration or to a software module. When adding a binding rule to a software module, you can set values for the following criteria:
   - A deployment scheme
   - A system profile
   - A current OS configuration
   - Administrative group
   - One of the system-definable and user-definable fields of the database (only used if you have customized the database)
   - An operating system type, such as Windows 2000
   - An operating system version, such as SP2
   - An operating system language
   - An operating system architecture, such as x86-32
   - A computer model name
   - A BIOS version
   - A PCI device
   - A base board
   - MultiChassi
   - HAL Type
   - A free-text condition in `Rembo-C;` syntax

   For example, to create a binding based on the operating system type between a software module and targets, you must create a new rule, click **OS type**, and select the operating system version that you want to limit this software module to.
2. When adding a binding rule to an OS configuration, you can set a condition on the deployment scheme, and on the computer model name. The next ten fields are only used if you have customized your database and want to match specific user categories.
3. Finally, you can enter a free-text condition following the `Rembo-C;` syntax. They must only be used by advanced users.

   The conditions determine the applicability of the rule and evaluate to `true` or `false`. A condition must be formed using the variables also used for keyword substitutions in software modules, combined with Java-like logical operators, listed by order of priority in the table:

Table 16. Logical operators for free-text conditions

Operator	Meaning
<	smaller than
<=	smaller than or equal to
=>	greater than or equal to
>	greater than
==	equal to
!=	not equal to
&&	AND operator
||	OR operator

For example, a typical condition can be:

```
Disk[0].DiskSize > 10*1024*1024
```

Note: If a condition cannot be evaluated, it is considered to have the value `false`.

Scheduling the application of software modules

It is not possible to schedule the application of software modules for this operating system and hardware.

All software modules are applied **When the OS is installed**, regardless of the set stage in the **Software application order** window. Reboots are not handled either.

Task templates for Solaris operating systems

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout

The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout

The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout

The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout

The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout

The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout

The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

Deployment schemes are not linked to architecture of the target or the operating system. Therefore, the deployment scheme wizard always offers to set all modifiable parameters. When deploying, parameters incompatible with either the architecture of the target or the operating system being deployed are not taken into account.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard, by clicking **New deployment scheme** from the **Task templates** page.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

The following parameters apply for simple one-time deployments and for redeployment operations.

Description

The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, therefore the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

When the deployment starts

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

Always edit target-specific parameters
to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

Edit parameters for targets that are not yet in the database
to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

Never edit parameters
to have unattended deployments if all BOMs have been previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

Yes, display a warning
if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment
if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, becauseTivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

Note: These parameters have no effect on PowerPC and Cell targets.

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of target computers to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed computer, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, /. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)

- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

Note: Multicast is available only if

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of computers that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast
to deploytargets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast, without explicit synchronization
to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.

Multicast, with synchronization on all targets
to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Multicast, with the following synchronization parameters
to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

You can use a network share or Linux HTTP protocol on the server to download the files to the target computers, rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP protocol provides a shorter operating system installation time. To use a network share or Linux HTTP protocol:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module). .

On-site deployment

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment parameters

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
Indicate if you want to keep the deployment image in a protected partition and the size of this partition.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation
This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user.

Unbind software module at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at

the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is currently relevant forWindows Vista/2008/7/2012deployments only.

Send mail at end
This option can be set only if a *sendmail* tunnel has been created.

Send mail to
This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network

Note: This parameter has no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button
This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Deploying Solaris

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

When deploying a Solaris system profile, you must have set the following target properties:

IP address
: To edit this field, go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Click **Switch to Advanced IP settings mode** in the **Common networking info** section.

Network mask
: To edit this field, go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Click **Switch to Advanced IP settings mode** in the **Common networking info** section.

Default Gateway

To edit this field, go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Click **Switch to Advanced IP settings mode** in the **Common networking info** section.

Name resolution method

To edit this field, go to **Server** > **OS deployment** > **Target Monitor**. Double-click on a target to view its details. Click **Edit** in the **UNIX-specific info** section. If you use DNS, then you must also set

- DNS server
- DNS domain
- DNS domain search order

in the **Common networking info** section.

Moreover, the target must already be registered in the DNS server. The target name (unqualified) entered in the OS deployment server database must match the DNS record.

If these requirements are not met, Jumpstart switches to interactive mode.

If you want to use DHCP, you must have set a DHCP reservation on your DHCP server to ensure a coherent name and IP address for your target during deployment.

To deploy Solaris system profiles, the OS deployment server must have access to the NFS share where the content of the installation CDs have been copied, because the content of the system profile is not stored on the OS deployment server. For the deployment, the NFS share can be on a Solaris target or they can have been moved to a UNIX target compatible with the Solaris targets

When you deploy a cloning system profile, you may encounter driver and/or disk mapping issues if the hardware of the reference target of the cloning system profile is significantly different from the hardware of the deployment target.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. Select **Simple deployment** and click **Next**
5. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Note: The Wake on LAN options are available only on Intel targets.

When the deployment is complete, the target boots into the operating system.

After deploying a Solaris unattended setup profile, you might see the following message:

```
sunablade0 console login: line 24: WARNING: loghost could not be resolved
```

This is standard Jumpstart behavior, allowing you to redirect loghost to a separate computer provided by DNS.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor:

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for Solaris:

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification:

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     - **No**
     - **Whatever the notification is**
     - **If deployment is successfully completed**
     - **If deployment failed**
  5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.
- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Provisioning AIX on PowerPC targets

To work with AIX system profiles on PowerPC targets, you must take into account some specificities of these targets.

Types of system profiles

Only unattended system profiles are supported on AIX. Cloning system profiles are not supported.

DHCP specificities

There are specific considerations for setting Dynamic Host Configuration Protocol (DHCP) options. Make sure you set them appropriately.

Note: Microsoft DHCP server does not work well with somePowerPC firmware. Use IBM recommended DHCP servers.

Registering new targets

You must add targets manually into the Target Monitor or import a comma-separated text file containing a list of targets to be added.

System profiles for AIX operating systems

A system profile is the partition layout and list of files to deploy an operating system.

The main purpose of Tivoli Provisioning Manager for Images is to deploy an operating system on targets by replicating a reference system. However, unattended installation of operating systems is also possible. The latter case Tivoli Provisioning Manager for Images does not replicate a reference system, but merely provides the correct parameters to the operating system setup for a fully unattended installation.

There are a number of differences between an unattended installation and disk cloning. First, creating an unattended installation in Tivoli Provisioning Manager for Images is straightforward. All of the necessary tasks are performed on the server, using the Web interface. In contrast, a cloning-mode system profile requires you to configure a target, prepare it for cloning, and run the cloning process directly on the target. However, the native mode of operation of the product is centered around cloning-mode system profiles, because this method of deployment is faster than unattended installation. When deploying computers on a large scale, unattended installation is not possible. Novice users might start with creating unattended installation profiles because this is easier than cloning-mode profiles.

Note:

- Disk cloning is not supported for Linux PowerPC and Cell targets. Only unattended setup is supported.
- To avoid failures in creating or deploying a system profile, clean up the temporary directory inside the OS deployment server installation directory on a regular basis.
- To create or deploy a system profile from a physical or virtual machine at least 512 GB RAM is required.

- In BIOS mode, system profiles can have a maximum of 3 primary partitions. Therefore, you cannot clone a target with 4 primary partitions.
- You cannot deploy Linux profile with an LVM root partition if you use deployment media.

Creating an unattended setup system profile for AIX operating system

An unattended setup system profile allows you to install operating systems using standard installation processes in unattended mode.

To create an AIX unattended system profile, you must work on an AIX operating system of the same version as the profile you want to create, and the web interface extension must be running.

Note: Using AIX NFS is not an easy process: if you have no DNS and no entry inside `/etc/hosts`, another computer can never mount the exported path. This may prevent a target to access the installation source during deployment. After profile creation, it is recommended to move the installation file to another NFS server.

1. Copy the AIX installation CD or DVD on the hard disk.
2. Export the path of the folder in which you have copied the installation files by NFS. This folder must have write permissions.
   a. Verify that NFS is already running by typing the command `lssrc -g nfs`. The output should indicate that the `nfsd` and the `rpc.mountd` daemons are active. If they are not, you must start the NFS daemons.
   b. At a command line, enter `smit mknfsexp`.
   c. Specify appropriate values in the fields
      - **Pathname of directory to export**
      - **Mode to export directory**
      - **Export directory now, system restart or both**
   d. Specify any other optional characteristics you want, or accept the default values by leaving the remaining fields as they are.
   e. When you have finished making your changes, SMIT updates the `/etc/exports` file. If the `/etc/exports` file does not exist, it is created.
   f. Repeat steps a through e for each directory you want to export.
3. Open the web interface, go to the menu and select **Profiles**.
4. Click **New Profile**. A system profile wizard opens to guide you through the steps of creating a profile.
5. Select **Unattended setup** in the first pane of the profile wizard.
6. Select your operating system from the list and click **Next**.
7. Follow the instructions of the wizard.

   When an AIX 6.11 profile is created, an `lpp` directory is created in the directory where the image was created. The `RPMS` and `installp` directories are moved to `lpp`, and the `usr/swlag` directory is moved to `lpp/usr`. The same principle applies also for AIX 5.3 system profiles. However, the directory names are not the same.

- When your first unattended installation profile is created, you can use it to deploy targets.
- If you want to create a new system profile, you must copy the AIX CD or DVD again, as indicated in step 1.

Note: When you create a new profile, use a directory different from the one of the previous profile, because the files in that directory are used during the deployment and if they are overridden the deployment can fail.

- Although all the files on the installation media were necessary for the profile creation, only some are needed at deployment time. If you want to delete superfluous files:

  for AIX 5L™ 5.3
  make sure *NFS_install* points to `install/ppc`. You can then delete the content of other installation directories.

  for AIX 6.11
  make sure *NFS_install* points to `lpp`. You must keep the `lpp` directory and all its sub directories, but you can delete the content of other installation directories.

- Because AIX NFS server is not easy to use, it is recommended to move the AIX installation content on a Linux or Solaris NFS server. You must also edit the OS configuration to update **NFS installation source** to the new value.

Organizing and editing system profiles

After you have created a system profile, you can view it on the OS deployment server through the web interface. The profiles are listed on **> Server > OS deployment > Profiles** page, in the **System profiles** pane. Each blue jacket represents a system profile (that is, the hard-disk partition images).

If you want to organize your system profiles, you can create subfolders by following these steps:

1. Select the parent folder with a left mouse click.
2. Call the contextual menu with a right mouse click.
3. Select the **Add a new profile folder** menu item.
4. Enter the new folder name.
5. Click **OK**.

You can then move profiles (by dragging-and-dropping the profile icons) from the top folder, where they are automatically created, to the appropriate subfolder.

Editing system profiles:
To display and edit the parameters associated with a given profile:

1. Double-click a system profile to open the **Profile details** page.
2. Click **Edit** on top of the parameter sections to edit the parameters.

Updating device mapping

You can update device mapping to force logical disks to point to specific physical devices.

Note: Updating device mapping from both the **Profile details** page and the **OS configuration details** page can lead to incorrect OS configurations and prevent OS deployment. Depending on your particular needs, choose one or the other entry point, and then perform all your changes from that entry point.

1. Go to **Server** > **OS deployment** > **System profiles** > **Profile details** or to **Server** > **OS deployment** > **System profiles** > **Profile details** > **OS configuration details**, **Disks** tab.
2. Click **Modify device mapping**.

3. Select to which physical device you want to map your logical disk. The column starting with **Disk 0** corresponds to an automatic detection of the first to the eighth disk, the column starting with **/dev/hda** corresponds to standard device names.

   Note: Spanning a logical disk on several physical disks is not currently possible.

   Note: On HP servers with smart array disk controllers, disk devices are usually named `/dev/cciss/cxdx`, where *x* is the disk number. Edit your device mappings accordingly.
4. Click **OK**.
5. Repeat step 2 on page 289 to step 4 for each logical disk for which you want to update device mapping.

If the new device mapping you selected is incorrect, you receive a warning message.

OS configurations and fixed common parameters

A system profile is the partition layout and list of files to deploy, while OS configurations are operating system parameters.

At the bottom of the **Profile details** page, there is a list of the OS configurations that correspond to your profile.

You can define several OS configurations for each system profile and duplicate them. These copies share the same image files, and the same partition layout, but can have different target parameters. You must then assign new values to some of the OS configurations parameters to make the original OS configuration and its copies unique.

If you want to automate the assignment of parameters to targets, you can view and edit the OS configuration you are about to deploy by clicking on its name in the **Profile details** page. You are now in the **OS configuration details** page. The information is divided into panels, each displaying sets of parameters. You can modify the parameters either through the web interface or by using a parameter file.

Editing OS configuration parameters in the web interface for AIX:

The web interface displays a number of OS configuration parameters divided into panes. You can edit these parameters in the web interface.

To edit parameters:
1. Click a tab to select the corresponding pane.
2. Click **Edit** on the banner of the section where you want to update parameters.
3. Modify the values.
4. Click **OK**.

Task templates for AIX operating systems

Task templates group together elements that can be customized on a target. These elements are mostly screen layouts, which condition the appearance of the target screen during the different phases of its control by Tivoli Provisioning Manager for Images.

A deployment scheme is a specific type of task template. Together with the target display screen layout, it contains other parameters for customizing a deployment on a target.

Note: Starting with Fix Pack, version 5.1.0.2, deployment schemes are considered to be a subset of task templates. The functions of deployment schemes have not been altered. To access deployment schemes, go to the **Task template** page, and select the **deployment scheme** folder.

The task template page of the web interface contains a task template tree in the left pane with seven folders. The content of the selected folder is displayed in the right pane.

There are seven task template folders in the tree. They are described here.

Deployment Schemes
Deployment schemes contain parameters that indicate how an OS configuration must be deployed on your target. The **deployment Schemes** folder contains at least the **Default** scheme.

Idle layout
The idle layout defines what is shown on the target when there is no pending task. The **Idle Layout** folder contains at least the **Idle state** layout.

Menu Layout
The menu layout defines how deployment menus are shown to the users. Menus are used when an OS configuration and on deployment CDs. The **Menu Layout** folder contains at least the **Menu** layout.

OS Detection Layout
The operation system detection layout defines the target display when a target is busy detecting the currently installed operating system. It is used when creating a cloning profile from the web interface. The **OS Detection Layout** folder contains at least the **Detect operating system** layout.

Profile Creation Layout
The system profile creation layout defines the target display when a target is busy creating a new system profile. It is used when creating a cloning profile from the web interface. The **Profile Creation Layout** folder contains at least the **Creating cloning profile** layout.

Profile Restoration Layout
The system profile restoration layout defines the target display during the manual restoration of a system profile by the administrator.

Note: A system profile restoration is always performed as-is and must not be confused with an automated deployment resulting in a fully configured operating system installation.
The **Profile Restoration Layout** folder contains at least the **Default OS Restoration** layout.

State Capture Layout
The state capture layout defines the target display when a target is saving the operating system state for future redeployments.

State Restoration Layout
The state restoration layout defines the target display when a target is redeploying an operating system from a saved state.

When a task template is selected in the right hand pane, the bottom of the web interface contains a link to **Customize GUI**. Follow this link to modify the look of your target screen . When the selected task template is a deployment scheme, there are additional links to view and edit the current scheme.

Creating and editing deployment schemes

By customizing your deployment schemes, you can adapt the way in which your predefined OS configurations are installed onto targets.

Deployment schemes are not linked to architecture of the target or the operating system. Therefore, the deployment scheme wizard always offers to set all modifiable parameters. When deploying, parameters incompatible with either the architecture of the target or the operating system being deployed are not taken into account.

1. The easiest way to create a new deployment scheme is to run the deployment scheme wizard, by clicking **New deployment scheme** from the **Task templates** page.
2. Alternatively, you can modify an existing scheme by editing its parameters. To do this, select a scheme and click **View deployment parameters** and then use **Edit** in the banner on top of each parameter section.
3. If you prefer using a wizard to edit your scheme, click **Edit parameters using a wizard**.

   The following parameters apply for simple one-time deployments and for redeployment operations.

   Description

   The first step is to enter a name for this deployment scheme. Make it explicit enough so that you can pick it easily when starting a deployment (the web interface does not show the settings in a deployment scheme, therefore the choice must be made by name only). Because deployment schemes determine how the computers are installed (and not what is being deployed), use a description such as `Multicast 50 targets` or `On-site` rather than the name of a OS configuration or of a group of computers.

   When the deployment starts

   Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

   You must determine if Tivoli Provisioning Manager for Images requires user interaction during deployment (to edit individualtarget parameters) or runs completely unattended.Select:

   Always edit target-specific parameters
   to have the opportunity to change the target parameters at every deployment. The parameter edition can be made either directly on the target computer or by double-clicking the target icon in the Target Monitor.

   Edit parameters for targets that are not yet in the database
   to be prompted only during the first installation of each target. Subsequent deployments for the same targets run unattended. This is the default choice.

   Never edit parameters
   to have unattended deployments if all BOMs have been

previously entered in the database. Any missing entry leads to a red banner on the target and cancels the deployment for this target.

You must also select how Tivoli Provisioning Manager for Images behaves when the model of the computer being deployed does not match the model of the computer on which the image was created. This feature requires DMI for hardware detection.Select:

No if you know that all your system profiles are fully hardware-independent or for deploying universal images.

Yes, display a warning
if you want to see all possible OS configurations for a computer, but want to avoid mistakes. This choice can require user interaction and is therefore not appropriate for a fully unattended deployment.

Yes, abort the deployment
if you want to prevent anyone from using an OS configuration on a computer different from the one for which it was designed.

Use 'BIOS fallback MBR' to start PXE is used when PXE activation (the process of enabling PXE when booting on the hard-disk) does not work.

The PXE boot code manages the multiple reboots needed to install a computer. To manage these reboots, the PXE boot code must intercept the boot process of the computer at every boot.

- If the computer is configured to always start on the network (LAN device first in the list of boot devices), there is nothing to do, becauseTivoli Provisioning Manager for Images is loaded into memory at every boot.
- If the computer is configured to start on the hard-disk, you can change the MBR of the hard-disk and make it point to the work partition at the end of the hard-disk. Tivoli Provisioning Manager for Images is then loaded from the hard-disk when the computer starts up, instead of loading the operating system. The disadvantage of this method is that, because the computer did not use the network card to boot, PXE is not available. To enable network access, PXE is activated with a special function in the PXE card that makes it behave as though the computer had booted on the LAN. However, this is not documented in PXE, and does not work on every network card. If the network does not support this, an error is raised, and access to the OS deployment server fails (the message **Network started**, followed by an error).

  When PXE activation does not work, you can write a special MBR telling the BIOS that the hard-disk is not a valid boot device. By default, the BIOS falls back to the next device in the list, which in most computers is the network. As a result, the computer boots on the network and has full access to the network. This is the purpose of the **Use 'BIOS fallback MBR' to start PXE** check box.

Data collection

Note: These parameters have no effect on PowerPC and Cell targets.

By default, Tivoli Provisioning Manager for Images automatically populates the database with an inventory of the hardware setups of all deployed computers. For Windows, a software inventory can also be populated based on the registry. If you are not interested in using those inventories, or if your computers do not comply with any of the hardware detection standards, you can disable these features. Be advised that running the hardware or software inventory on thousands of computers can produce a huge database. This inventory is performed on locked screen.

Tivoli Provisioning Manager for Images centrally reports the status of the deployment of target computers to the OS deployment server and to the server database. Additionally, if you want to keep the deployment logs and the list of software modules on each deployed computer, you can specify a local path where the log files are to be stored. The path that you specify is relative to the root of the operating system on the target, for example, /. In the deployment scheme details, the label of this field is **Save deployment log to**.

When the deployment is completed

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

When the deployment process is finished, you can select if you want to:

- Turn off the computer automatically (if supported)
- Boot the operating system automatically (this value might not make sense with some values of the previous setting)
- Display a green banner and wait for a manual shutdown

Network usage

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

Note: Multicast is available only if

- The targets have an Intel x86 or x86-64 architecture
- Multicast is selected in the deployment scheme
- The subnet supports multicast traffic
- Multicast is not disabled in the boot options of the target
- The target is not a VMWare 3.0 guest.

Depending on the number of computers that you are deploying simultaneously on your infrastructure, you must select one of the following networking modes:Select:

Unicast
to deploytargets one by one, or if you cannot use multicast. When deploying several targets simultaneously in unicast, the deployment time increases dramatically with the number of targets, as the result of network saturation.

Multicast, without explicit synchronization
to use soft-synchronized multicast protocol. Using this protocol, every target independently starts downloading images at soon as it is ready, and continues with the deployment as soon as it has downloaded all required material. When two or more

targets (using the same deployment scheme) are downloading files in parallel, they automatically share the same bandwidth. The fastest target has the priority for the choice of the next shared files to be sent by the server, but the slower targets can receive them if they need them. This is a scalable solution that allows for a rolling deployment scenario.

Multicast, with synchronization on all targets

to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameter which is the maximum timeout before starting in any case. Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Multicast, with the following synchronization parameters

to use a classical replicated multicast method. This mode is adequate for installing computers in batches. Enter the replication parameters (for example, the number of targets to wait for before starting the download, and the maximum timeout before starting in any case). Tivoli Provisioning Manager for Images multicast protocol can accept new download targets even after the initial replication period is over, and integrate them seamlessly into the transfer.

Note: In the first stage of an OS deployment, there are two target synchronization stages. Therefore it might seem that the maximum timeout that was set before starting the deployment is doubled.

You can use a network share or Linux HTTP protocol on the server to download the files to the target computers, rather than downloading the whole image to the hard disk of the target. Using a network share or Linux HTTP protocol provides a shorter operating system installation time. To use a network share or Linux HTTP protocol:

- Select **Download files with a network share or Linux HTTP when applicable** in the deployment scheme.
- Share the `files\global\partition` directory and provide at least read-only access to it.
- Enter the relevant server parameter. Go to **Server** > **Server parameters** > **Configuration**. (See Network share module). .

On-site deployment

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

If you are running a one-time deployment in a deployment center and do not want to use redeployment, leave the check box blank and click **Next**.

If you are running an on-site deployment, or if you plan to use redeployment, you can enable the advanced feature.

Redeployment parameters

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.

Indicate if you want to keep the deployment image in a protected partition and the size of this partition.

Note: The following parameters cannot be modified using the wizard. You must edit your deployment scheme parameters.

Request user confirmation

This parameters allows you to ask for user confirmation before running a deployment.

Unbind OS configuration at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the OS configuration from the target at the end of the deployment. This OS configuration is not proposed the next time the target boots and, if no other OS configuration is bound, the target presents a locked screen to the user.

Unbind software module at the end

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `No` by default. Setting this parameter to `Yes` unbinds the software module at the end of the deployment. This software module is not proposed and installed the next time a deployment is performed.

Vista 2008 Windows 7 **Disable user interaction during deployment**

Note: These parameters have no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **General settings** section, is set to `Yes` by default. If you set this parameter to `No`, you can obtain a command prompt by pressing Shift-F10 on the target computer during a deployment to modify deployment files.

Note: This parameter is currently relevant forWindows Vista/2008/7/2012deployments only.

Send mail at end

This option can be set only if a *sendmail* tunnel has been created.

Send mail to

This parameter is available only if **Send mail at end** is not set to **No**. Use this field to enter the e-mail address to which information must be sent at the end of the deployment.

Configure Network

Note: This parameter has no effect on PowerPC, Cell, and SPARC targets.
This parameter, located in the **Network settings** section, is set to **Before software installation** by default. The network setting of the target are set before software modules are installed, enabling the product to use the network settings during the installation of the software modules. Set the parameter to **After software installation** if you want the network settings to be applied after software modules are installed, for

example if you intend to physically move the target after deployment and want it to be configured with the network settings for its final location.

Disable cancel button
This parameter, located in the **Client Display** section, allows you to prevent users from aborting a deployment by removing the cancel button. Set the parameter to **Yes** if you want to remove the cancel button from the client display

You can use the newly created deployment scheme to deploy a system profile.

To delete a deployment scheme, select the scheme, then right-click it, and click **Delete**.

Deploying AIX

A deployment is the process of installing an operating system on a target, and configuring the operating system for a specific user.

Deployment requirements

To start a deployment on a target, several elements must be present in the database.

The following elements are required:

- A *deployment* scheme associated with the target to deploy. The deployment scheme determines how to deploy the operating system on the target. If there is no association between a deployment scheme and the target to deploy, Tivoli Provisioning Manager for Images automatically uses the **Default** deployment scheme.
- An operating system *configuration* that is used to select which operating system to install. If there is no OS configuration associated with the target to deploy, the deployment does not start.
- Optional *software modules* to install in addition to the operating system during the deployment process. If there is no software module associated with the target to deploy, the operating system image is deployed without modification.

The OS configuration and the software modules can be considered to be the *content* of the deployment. A deployment scheme is the *method* of deployment.

The database keeps information about associations (*bindings*) between targets and deployment schemes, between targets and OS configurations, and between targets and software modules. These bindings can be configured manually or with binding rules (for example, `deploy configuration windows XP on targets whose model name starts with` *`Dell`*).

The minimal binding required to start a deployment is an OS configuration. If no configuration is bound to a target, the deployment does not start. In practice, however, Tivoli Provisioning Manager for Images always asks for an OS configuration and deployment scheme when beginning a deployment.

Tools to start and configure deployments

Bindings between targets and deployment elements are necessary to perform a deployment. You can create and edit these bindings in the **OS configurations** panel of the **Target details** page.

The Target Monitor provides functions to prepare a deployment, start a deployment, follow the progress of a deployment, and organize targets.

Binding rules are used to create permanent implicit bindings between targets and deployment elements, without explicitly creating the binding for each target. They are created with the web interface. OS configurations and software modules contain a specific section at the bottom of the **Details** page for creating automatic binding rules.

Starting a one-time deployment

You start deployments in the web interface by indicating on which target or targets the deployment must occur.

When you deploy a cloning system profile, you may encounter driver and/or disk mapping issues if the hardware of the reference target of the cloning system profile is significantly different from the hardware of the deployment target.

To start a deployment:

1. Select a single target or multiple targets on the Target Monitor page. To select multiple targets or deployment, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the Ctrl key.
2. Select **Deploy now** in the contextual menu.
3. In the first screen of the deployment wizard, you can choose to use the same deployment parameters as the previous deployment.
4. Select **Simple deployment** and click **Next**
5. Follow the deployment wizard instructions to select a deployment scheme, an OS configuration and optionally software modules, and to set up deployment options.

   Note: The Wake on LAN options are available only on Intel targets.

When the deployment is complete, the server either displays a green banner on the target, boots in the operating system, or powers the target off, depending on how the deployment scheme is configured.

After deployment, the following warning message might appear in the logs and can be safely discarded. `Warning: partition x does not end at a cylinder boundary`. Partitions are aligned on megabytes rather than on cylinders. Aligning on megabytes is recommended by virtualization companies because it is safer when you deploy on both physical targets and virtual machines. The only drawback is an incompatibility with DOS.

Monitoring deployments

There are several ways available to monitor the deployment progress.

Monitoring deployment progress with the Target Monitor:

You can use the Target Monitor to monitor deployments remotely. Information is located on the **Target Monitor** page and on several tabs of the **Target details** page.

Note: The following rules apply to the **Expire** field located on the **Target Monitor** page:

This date, if not explicitly set, for example if specified for tasks launched with Java API, is calculated as follows:

- ExpireDate = DoneDate + ExpirationTimeout
- DoneDate = ExecDate + DoneTimeout

where DoneDate is the date after which the targets should not start performing the activity, but can continue if they already started performing it.

DoneTimeout is, by default, 2 days. ExpirationTimeout is, by default, 14 days. If you want to change these timeout values, you can add in the `config.csv` file for each server the related keywords with its values (specified in days).

Sample config.csv:

```
HostName;DoneTimeout;ExpirationTimeout
serv-01;5;30
```

to set DoneDate to 5 days after ExecDate, and ExpireDate to 35 days after ExecDate.

The information in config.csv will be loaded at restart of the server, or can be reloaded, while the server is running, using the `rbagent rad-configure` command. The customized values affect the new activities and the deletion of existing and completed activities.

On the **Target Monitor** page, the target color changes during the deployment. When PXE is activated, targets are monitored on a regular basis. The color of the icon is updated as soon as the status changes. By pointing to the target icon, you can get a description of the target status.

Note: A successfully deployed computer can continue to have a yellow icon (indicating that the deployment is still in progress). This reflects a PXE activation problem. The computer, having booted on the hard disk, is not using the network to inform the OS deployment server of its status. To remedy this, select the **Use 'BIOS fallback MBR' to start PXE** check box in the deployment scheme wizard. This forces the computer to boot through the network first.

If the deployment scheme used is configured to collect inventory information about target hardware (which is the default), you can see information about target hardware in the **Inventory** panel of the **Target details** page for that target (double-click on the target to go to the details page).

At the end of the deployment, the target icon shows either a green screen (success), or a red screen (failure). The deployment logs stored on the OS deployment server provide information about the deployment process. They are particularly useful in case of deployment failure to track its cause. To access the logs, double-click the wanted target. This opens the **Target details** page. Select the **Logs** tab to display a list of logs. To view a specific log, click its description. To download it, click **download** immediately after the log description.

Note: Logs are text files with UTF-8 formatting. If you are using a Windows operating system, you can view log files adequately by opening them in Microsoft WordPad.

There is only one log file for each deployment. This log file contains information about the different stages of the deployment process, including reboots and information provided by the operating system being deployed.

If any log information needs to be propagated to the OS deployment server outside of any task, an *idle* log file is created to store this information. The idle log file is created on demand and does not therefore exist for all the targets.

Another place of interest for information about a current deployment or another current task is the **Task history** tab, where each task of the target is listed. For each current task, the following information is provided:

- Description
- Status
- Scheduled date
- Start date
- Progress rate
- End date
- Download link to the log file
- Download link to the task file
- Download link to the `bom` file

Note: You must scroll to the right of the **Task history** tab to see all the fields.

The log file contains the target log. The task file contains all the parameters of the task. The `bom` file contains target-specific parameters for the given task.

The log file, the task file, and the `bom` file are needed by the development team to fix defects. Make sure that you download these three files if you suspect the presence of a defect in the software.

To cancel or destroy a task, select the task and select **Cancel target task** or **Destroy target task** in the contextual menu. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on running tasks, because the task destruction can cause strange behavior.

To view tasks scheduled for a later time, go to the **Tasks** page.

Monitoring deployment progress with the Tasks page for AIX:

The **Tasks** page is also a useful source of information to monitor a deployment (and other tasks). You can also cancel tasks from there.

Go to **Server** > **Server history** > **Tasks**. .

The description field of each deployment in the **Tasks** page is headed by the keyword *Deploy* for easy retrieval. The information provided includes

Description
Is headed by specific keywords, indicating the type of task. *Deploy* is the keyword for deployment tasks.

Execution
Is the scheduled date and time for the execution of the task.

State
Uses icons to represent if the task is pending, in progress, completed, and so on. If in doubt to the meaning of a state icon, browse over it to get a state name.

Progress
Indicates the rate of completion of the task as a percentage.

Expire Indicates when the task information is going to be removed from the page.

Tasks are expandable by clicking their **+** sign. An expanded task displays information about its targets. The target information fields are:

- IP address
- Hostname
- Start date and time of the task
- State
- Progress rate
- Status date

If, for any reason, you want to cancel a running or scheduled task, you can easily do so by following these steps:

1. Expand the task
2. Select the target for which you want to cancel the task
3. Select **Cancel task** from the contextual menu
4. It is also possible to *destroy* tasks. When you destroy a task, all its records and files are permanently deleted. Use this option with caution, especially on a running task, because its destruction can cause strange behavior. To permanently delete tasks:
   a. Select one or several tasks. To select multiple tasks, use the Shift key for a range of tasks and the Ctrl key for individual tasks.
   b. Select **Destroy task** from the contextual menu

Receiving an e-mail notification:

To receive an e-mail notification at the end of a deployment, you must configure a TCP tunnel called *sendmail*.

To receive an e-mail notification at the end of a deployment, you must have configured a *sendmail* TCP tunnel.

Note: The OS deployment server supports only sendmail servers without authentication.

There are two options to configure a deployment to receive an e-mail notification:

- You can edit the deployment scheme used for deployment to include the notification information.
  1. Go to **Server** > **OS deployment** > **Task Templates**.
  2. Select **Deployment schemes** and double-click a specific deployment scheme name to edit it.
  3. Click **Edit** in the **General settings** section.
  4. Under **Send mail at end:**, select the type of notification that you want. You can choose among:
     - **No**
     - **Whatever the notification is**
     - **If deployment is successfully completed**
     - **If deployment failed**

5. If you selected a notification, you must now enter a valid e-mail address to which the notification is sent, under **Send mail to:**. You can enter only one e-mail address. If you want to send the notification to several addresses, use a distribution list.

- You can modify the settings of the deployment scheme in the deployment wizard. Step 4 and possibly step 5 are available.

Depending on your selection, you will receive an e-mail notification at the end of the deployment.

The notification e-mail looks like this:

The target with MAC *xxxx* / IP **xxx** has completed an activity *activity description*.

You can now deploy targets with the edited deployment scheme and receive e-mail notification at the end of the task.

Creating a sendmail TCP tunnel:

A *sendmail* TCP tunnel is mandatory to receive e-mail notification at the end of a deployment.

Note: OS deployment server supports only sendmail servers without authentication.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New tunnel**.
3. In the TCP tunnel information screen enter,
   a. The name of the tunnel. In this case the name is `sendmail`.
   b. The host name or IP address of the SMTP server.
   c. The TCP port of the SMTP server.
4. Click **Save**.

You can now configure deployment schemes to send an e-mail notification at the end of a deployment.

Chapter 7. Managing virtual images

With the Image Monitor feature, you can capture images, copy and customize them as you need, and then deploy them to physical or virtual machines. To the physical machines apply the same requirements of system profiles.

Virtual images are physical machines, hypervisors, VM instances, and snapshot images.

The virtual machines that are supported are on VMWare ESX, Hyper-V, Xen, and Kernel-based Virtual Machine (KVM).

After the images are deployed, you can manage them by monitoring, updating, and tracking changes.

It is possible to deploy or direct migrate a virtual image from a BIOS machine to a UEFI machine. During this operation, ESP and MSR are added on the fly and patched accordingly.

Using the Image Monitor feature, you can also convert an image from one format to another format, to address server workload or application environment problems. This feature can be found in the Web interface by navigating the following path: **Advanced features** > **Image monitor**

- If you need to migrate a physical machine due to hardware constraints, you can convert it into a virtual image. You can then deploy the hardware as a hypervisor and install the virtual image as a guest on this new hypervisor. For example, if you need to build or consolidate a new data center, or increase its capacity, you can move the workload from the data center to virtual machines by converting the system profile datacenter into the native format of a hypervisor (such as VHD, VMDK). You can then deploy the virtual image to a virtual machine.
- If, after testing an application on a virtual machine, you need to move it into production on a physical server, you can convert its virtual image to a physical image and deploy it.
- You might also migrate servers running on one kind of virtual machine to a different kind of virtual machine, for example, servers that were developed and tested in Microsoft Virtual Server 2005 might be migrated to VMWare machines running on a VMWare ESX Server. In this case you can use the Image Monitor feature to migrate the servers by converting their current virtual image to another virtual image.

Hypervisor and guest requirements

This section contains the information you need to discover, capture and deploy virtual images. Before you use this feature, ensure that you meet all installation requirements on the hypervisor and guest environment.

Hypervisor

The hypervisor also called virtual machine monitor (VMM), allows multiple operating systems to run concurrently on a host computer. The hypervisor presents the guest operating systems with a virtual platform and monitors the execution of the guest operating systems.

Guest or virtual operating system

A guest or virtual operating system is the environment where a virtual machine runs.

Hypervisor requirements

This section contains the requirements of a hypervisor to discover, capture, and deploy virtual images.

Hypervisor types

Tivoli Provisioning Manager for Images works with the following types of hypervisor:

- VMWare ESX
- KVM
- XEN
- Hyper-V

Table 17 presents the hypervisor versions and the operations that can be performed with the product. An *X* indicates that the feature is supported by the product. A blank indicates a feature not supported by the product for the specific hypervisor.

Table 17. Hypervisor versions and supported operations

Hypervisors	PXE deployments	Boot on OS	Image capture and deploy	OVF import and export	Direct migration from a virtual machine	Network boot CD/ DVD	Offline deployment media
VMWare ESX 3.5	X	X	X	X	X	X	X
VMWare ESX 4.0	X	X	X	X	X	X	X
VMWare ESX 4.0 U1	X	X	X	X	X	X	X
VMWare ESX 4.0 U2	X	X	X	X	X	X	X
VMWare ESX 4.0 U3	X	X	X	X	X	X	X
VMWare ESX 4.1	X	X	X	X	X	X	X
RHEL KVM 5.5		X	X	X	X	X	X
RHEL KVM 5.6		X	X	X	X	X	X
RHEL KVM 5.7		X	X	X	X	X	X
RHEL KVM 6.0	X(*)	X	X	X	X	X	X

Table 17. Hypervisor versions and supported operations (continued)

Hypervisors	PXE deployments	Boot on OS	Image capture and deploy	OVF import and export	Direct migration from a virtual machine	Network boot CD/ DVD	Offline deployment media
RHEL KVM 6.1	X(*)	X	X	X	X	X	X
Citrix XEN 5.5	X(*)		X(*)	X	X(*)		
Citrix XEN 5.6	X(*)		X(*)	X	X(*)		
Microsoft Hyper-V 2008 R2	X		X	X	X	X	X
Microsoft Hyper-V 2008 R2 SP1	X		X	X	X	X	X

(*): You can run the operation in kernel-free mode only.

VMWare ESX 3.5 hypervisor requirements

To successfully discover, capture, or deploy a virtual image on VMWare ESX 3.5 hypervisor, you must install the Virtual Disk Development Kit (VDDK) Version 1.1 library on the hypervisor.

To configure the VMWare ESX 3.5 hypervisor:

1. Configure the active firewall or disable it by running, for example, the following command: `etc/init.d/firewall stop`.
2. Download the FUSE 2.5.3 library from SourceForge.net and compile it on the VMWare ESX hypervisor. For details see the `INSTALL` instruction file in the uncompressed library folder. This library is a prerequisite of VDDK.
3. Download VDDK version 1.1 from VMWare Cloud Computing with Virtualization and install it on a VMWare ESX hypervisor.
4. Define the VDDK path in the library path of the user running the rbagent by adding the following line to the `/root/.bashrc` file and reload it: `export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib/vmware-vix-disklib/lib32/`
5. Install libxml2 version 2.6.16 or copy the `libxml2.so.2.6.16` file and create the following symbolic link:
   ```
   ln -s /usr/lib/libxml2.so.2.6.16 /usr/lib/libxml2.so.2
   ```
6. After the VDDK installation, reboot the hypervisor.
7. Ensure that the VMWare ESX hypervisor supports the following command:
   ```
   vmware-cmd -l
   ```
8. Before running the full inventory, capture, or deployment of a virtual machine, ensure that the virtual machine is powered off.
9. Ensure that the web interface extension is installed and has the following content:
   ```
   ESX_HOST=127.0.0.1
   ESX_USER=root
   ESX_PWD=password
   ESX_PORT=902
   ```

10. Ensure that the vmware.conf file is located in the same directory where the web interface extension is installed and has the following content:

```
ESX_HOST=127.0.0.1
ESX_USER=root
ESX_PWD=password
ESX_PORT=902
```

The following is an example of a running VMWare ESX hypervisor configuration:

- VMWare ESX 3.5 U4
- fuse-2.5.3.tar.gz (compiled by running the **configure**, **make**, and **make install** commands)
- VDDK library: VMware-vix-disklib-1.1.0-163495.i386.tar.gz
- /usr/lib/vmware-vix-disklib/lib32 defined in the library path
- libxml2 version 2.6.16 (with libxml2-2.6.16-1.1.el3.rf.i386.rpm and libxml2-python-2.6.16-1.1.el3.rf.i386.rpm) or a link between /usr/lib/libxml2.so.2 and /usr/lib/libxml2.so.2.6.16 (ln -s /usr/lib/libxml2.so.2.6.16 /usr/lib/libxml2.so.2)

Installing VDDK

To work with the VMDK disk image format (import/export OVF format), you need VDDK 1.1 installed on the machine where the web interface extension or the OS deployment server is running.

When installing VDDK, change the environment variables as follows:

1. On Windows systems, add the binary path (such as C:\Program Files\VMware\VMware Virtual Disk Development Kit\bin) to the %PATH% variable.

   On Linux systems, add the library installation path (such as /usr/lib/vmware-vix-disklib/lib32) to the operating system library search paths.
2. Refresh the environment variables. If needed, restart your machine.

VMWare ESX 4.x hypervisor requirements

To successfully discover, capture, or deploy a virtual image on VMWare ESX 4.x hypervisor, you must install the Virtual Disk Development Kit (VDDK) Version 1.1 or 1.1.1 library on the hypervisor.

To configure the VMWare ESX 4.x hypervisor:

1. Ensure that VMWare ESX 4.x is installed by running cat /etc/vmware-release.
2. Install a supported FUSE library. For example you can download an already compiled libfuse RPM package such as libfuse-2.7.4-1.el5.pp.i386.rpm fromhttp://rpm.pbone.net/index.php3/stat/4/idpl/9551006/com/libfuse-2.7.4-1.el5.pp.i386.rpm.html. Install FUSE on the hypervisor by entering the following command:

   ```
   rpm -ivh libfuse-2.7.4-1.el5.pp.i386.rpm
   ```
3. Install VDDK 1.1 or VDDK 1.1.1. Available packages are VMware-vix-disklib.1.1.0-163495.i386.tar.gz (32-bit version) and VMware-vix-disklib-1.1.1-207031.i386.tar.gz.

Run gunzip, untar -xvf to extract the content and then ./vmware-install.pl to install the library. A warning message about a missing libfuse.so.2 library might be displayed even if the library is correctly installed. Proceed with the next configuration steps.

4. Do not install libxml2 because a supported version is already installed in VMWare ESX 4.0. Check that libxml2-2.6.26 is already present.
5. On VMWare ESX 4.0:

   Export LD_LIBRARY_PATH=:/usr/lib/vmware-vix-disklib/lib32 in the shell where rbagent runs.

   On VMWare ESX 4.1:

   a. Edit the file /etc/ld.so.conf and add /usr/lib/vmware-vix-disklib/lib32
   b. Refresh the current shell settings with the **ldconfig** command.
6. Create vmware.conf. Ensure that the vmware.conf file is located in the same directory where the web interface extension is installed and has the following content:

   ```
   ESX_HOST=127.0.0.1
   ESX_USER=root
   ESX_PWD=password
   ESX_PORT=902
   ```
7. Configure the active firewall to make rbagent able to contact OS deployment server
8. After the VDDK installation, you might need to reboot the hypervisor.
9. Ensure that the VMWare ESX hypervisor supports the following command:

   ```
   vmware-cmd -l
   ```
10. Run the web interface extension on the VMWare ESX hypervisor with the rad-refreshhypervisor option. To do this on Linux, copy rbagent.linux from the <TPMfOSD_DATA_DIR>\global\http\agents directory of the OS deployment server to the VMWare ESX hypervisor, set the execution permission: chmod +x rbagent.linux, and launch the web interface extension with the following syntax:

    ```
    ./rbagent.linux -d -s <server ip>:<Web_user_interface_administrator_password>
    rad-refreshhypervisor
    ```

    Ensure that only one instance of web interface extension is running.

The following is an example of a running VMWare ESX hypervisor configuration:

- VMWare ESX 4.0
- libfuse-2.7.4-1.el5.pp.i386.rpm
- VDDK library: VMware-vix-disklib.1.1.0-163495.i386.tar.gz
- /usr/lib/vmware-vix-disklib/lib32 defined in the library path
- libxml2 version 2.6.26

KVM hypervisor requirements

To successfully discover, export, import, or deploy a virtual image on a KVM hypervisor, you must ensure that the libvirt library is installed and configured on the hypervisor.

Note: You can run an image deployment on the KVM hypervisor, if you boot the KVM guest in a kernel-free mode or if you use a network boot CD or DVD.

Red Hat **6.x** If the KVM hypervisor is installed on RedHat 6.x, the 64-bit version of the web interface extension is required, because on this version only 64-bit libraries are available for virtual image detection.

Before starting a hypervisor discovery, the web interface extension checks if the libvirt library is installed otherwise the discovery mechanism uses the `virsh` commands on each hypervisor.

To ensure that the libvirt library is correctly installed, check that the `libvirt.so` file, used by the web interface extension to detect libvirt installation, is defined in the hypervisor library path. If the `libvirt.so` file does not exist, you must create it as a symbolic link pointing to the libvirt library file.

For example, you can enter the following command:

Red Hat **5.x**

```
ln -s /usr/lib/libvirt.so.0 /usr/lib/libvirt.so
```

Red Hat **6.x**

```
ln -s /usr/lib64/libvirt.so.0 /usr/lib64/libvirt.so
```

In this way `libvirt.so` points to the `libvirt.so.0` libvirt library.

KVM native commands: If you want to use the `virsh` commands, ensure that the following command:

```
capabilities
```

runs correctly from a virsh shell.

KVM discovery works with objects defined in libvrt: The web interface extension runs the KVM discovery only on objects that are defined in the libvirt library.

KVM requirements: KVM hypervisor machine must be 64 bit.

KVM hypervisor must have CPU with virtualization extension (check for `vmx`, `svm` flags under `/proc/cpuinfo` as follows: `egrep '(vmx|svm)' /proc/cpuinfo`)

Additional requirements: Before running the full inventory, capture, or deployment of a virtual machine, ensure that the virtual machine is powered off.

When creating the virtual machine select `KVM` as hypervisor and not `QEMU`. (To verify that the virtual machine runs on a KVM hypervisor, you can run the following command: `virsh dumpxml <kvm_guest_name>`, the output must contain the following information `<domain type='kvm' id='xx'>`)

Ensure that the web interface extension is running on the KVM hypervisor with the `rad-refreshhypervisor` option. To do this on Linux:

Red Hat **5.x**

Copy `rbagent.linux` from the `<TPMfOSD_DATA_DIR>\global\http\agents` directory of the OS deployment server to the KVM hypervisor, set the execution permission: `chmod +x rbagent.linux`, and launch the web interface extension with the following syntax:

```
./rbagent.linux -d -s <server ip>:<Web_user_interface_
administrator_password> rad-refreshhypervisor
```

Red Hat **6.x**

Copy `rbagent.lnux64` from the `<TPMfOSD_DATA_DIR>\global\http\agents` directory of the OS deployment server to the KVM hypervisor, set the execution permission: `chmod +x rbagent.lnux64`, and launch the web interface extension with the following syntax:

```
 ./rbagent.lnux64 -d -s <server ip>:<Web_user_interface_
administrator_password> rad-refreshhypervisor
```

Ensure that only one instance of web interface extension is running.

The raw disk is the only virtual disk format supported.

XEN hypervisor requirements

To successfully discover, export, import, or deploy a virtual image on a XEN hypervisor, you must ensure that the virtual machine is powered off.

Note: The operations supported on the XEN environment are the following:

- Discovering existing virtual machines and virtual images
- Exporting the discovered virtual images to OVF format
- Migrating virtual images into your image repository stored on the OS deployment server
- Deploying virtual images stored in the hypervisor to another machine using the **Deploy Now** > **A virtual image (system snapshot)** function.

Ensure that the XEN hypervisor meets the following requirements:

- It is a 64-bit machine.
- A web interface extension is running on the XEN hypervisor with the `rad-refreshhypervisor` option. To do this on Linux, copy `rbagent.linux` from the `<TPMfOSD_DATA_DIR>\global\http\agents` directory of the OS deployment server to the XEN hypervisor, set the execution permission: `chmod +x rbagent.linux`, and launch the web interface extension with the following syntax:

  ```
  ./rbagent.linux -d -s <server ip>:<Web_user_interface_administrator_password>
  rad-refreshhypervisor
  ```
- Only one instance of web interface extension is running.

The VHD disk is the only virtual disk format supported.

The supported storage repositories on XEN 5.5 are LVM, EXT, and NFS.

Guest requirements

This section contains the requirements of guest systems.

Supported guest systems

Tivoli Provisioning Manager for Images works with the following guest systems:

- KVM
- XEN
- VMWare
- Hyper-V

Supported file systems

- Tivoli Provisioning Manager for Images does not support the capture or deployment when the file systems are ReiserFS and exFAT.
- With LVM, the product supports only one volume group per disk and a volume group cannot span over two disks.

Prerequisites for KVM guests

Before deploying images on KVM guest, ensure you have the following configuration.

To deploy images on KVM guests:

1. Ensure KVM guests run on a KVM hypervisor.
2. Set the KVM guest boot device to `CDROM` and use a network boot CD/DVD or set the KVM guest boot to kernel-free mode.
3. Set the KVM guest as follows.

   Guest operating system

   Always set the guest operating system to Windows 2008 or Windows Vista even when you deploy a different Windows operating system (such as Windows 2000, Windows 2003, Windows XP, or Windows 2007).

   This is because if you create KVM guests of type Windows XP or Windows 2000, the KVM guest is created without the ACPI features needed to load the WinPE deployment engine. During the boot sequence the following error message is displayed: `Windows failed to load because the firmware (BIOS) is not ACPI compatible`. If you run the command `virsh dumpxml <machine_name>` the `<acpi/>` tag is not created for this machine type. Even though you can manually add the missing tag with the `virsh edit <machine_name>` command, the recommended method is to create the KVM guests by setting Windows 2008 or Windows Vista as guest operating systems.

   Note: Ensure that KVM Windows guests have minimum 1 GB of RAM to load WinPE3.

   Network adapter

   The KVM guest must use an Intel e1000 network adapter.

   Virtual disk controller

   The KVM version provided with RHEL 5.4 kernel (2.6.18) does not provide the virtual SCSI controller support. For this reason only KVM guests with virtual IDE controller are supported.

   Virtual disks and physical devices

   KVM guests support virtual disks using physical block devices but do not support physical host devices.

   You can add a virtual disk to a KVM guest by selecting a physical block device (`/dev/sdb`, `/dev/LVM`) on your hypervisor host from the Virtual Machine Manager as follows: **Add Hardware -> Hardware type: Storage ->** `block device (partition)`.

   You cannot add a virtual disk to a KVM guest by selecting a physical host device on your hypervisor host from the Virtual Machine Manager as follows: **Add Hardware -> Hardware type: physical host device**

Prerequisites for XEN guest systems

To manage Linux images on XEN, you must set some prerequisites on the Linux XEN guest system.

To work with a Linux guest system on XEN hypervisors, you must create it as a fully virtualized machine or convert the para-virtualized machine into a fully virtualized one, as in the following example::

1. Start the Linux XEN guest system.
2. Run `uname -a` to check if you are running a para-virtualized or a fully-virtualized machine.
   - Para-virtualized machine has a XEN kernel:
     ```
     Linux localhost.localdomain 2.6.18-128.el5xen
     #1 SMP Wed Dec 17 12:01:40 EST 2008 x86_64 x86_64 x86_64 GNU/Linux
     ```
   - Fully-virtualized machine does not have a XEN kernel:
     ```
     Linux localhost.localdomain 2.6.18-128.el5
     #1 SMP Wed Dec 17 12:01:40 EST 2008 x86_64 x86_64 x86_64 GNU/Linux
     ```

   If you are running a fully-virtualized machine, you do not need to perform any configuration step.
3. If you are running a para-virtualized machine you must perform the following steps:
   a. Install the non-XEN kernel using RPM.
   b. Edit `grub.conf` to boot from the new kernel by checking that `default` field points to non-XEN kernel.
   c. Remove the `"co" getty` service in the `/etc/inittab` file to restore standard gettys services as follows:
      - Remove line `co:....agetty`
      - Uncomment `mingetty to ttyN terminals`
   d. Stop the Linux XEN guest.
   e. Change its configuration parameters as follows:
      ```
      xe vm-list params=all uuid=uuidd_of_machine_you_are_converting
      xe vm-param-set uuid=uuidd_of_machine_you_are_converting
      HVM-boot-policy="BIOS order"
      xe vm-param-set uuid=uuidd_of_machine_you_are_converting
      HVM-boot-params:"order=dc"
      xe vm-param-clear uuid=uuidd_of_machine_you_are_converting
      param-name=PV-bootloader
      xe vm-param-clear uuid=uuidd_of_machine_you_are_converting
      param-name=PV-args
      ```
   f. Restart the Linux XEN guest and check that it works in fully-virtualized mode.
   g. You can now rediscover and manage the Linux XEN guest system with Tivoli Provisioning Manager for Images.

Prerequisites for Hyper-V guests

To be able to perform tasks on Hyper-V guests from the OS deployment server using PXE-Boot, you must modify the standard guest machine.

You must enable the Hyper-V guest to boot on the network.

1. Replace the default network interface controller **Network Adapter** with **Legacy Network Adapter** to enable the guest to boot on the network.
2. Change the boot order. Set the network interface controller **Legacy Network Adapter** as first device.

Note: After completing the tasks requiring connection to the OS deployment server, it is recommended to change the boot order again. Set the hard disk (IDE) as first device. This prevents the guest from booting into the OS deployment server instead of on the operating system.

Capturing images and storing on the server

You can capture an image of systems such as computers, physical servers, hypervisors, and virtual machines and store it on the Tivoli Provisioning Manager for Images server.

To capture the image of a reference target running a Linux or a Windows Vista/2008/7/2012/8 operating system, the reference target must have at least 1GB of RAM.

Windows If you capture a Windows virtual machine, ensure that you bound the required disk and network drivers in the WinPE deployment engine. For more details about binding drivers, see "Binding drivers to a Windows image" on page 315.

Note: If both WinPE 3.x and WinPE 4.x are available, WinPE 4.x is used. If target does not support WinPE 4.x, to run this task ensure that model patterns for WinPE 4.x do not match the target model. WinPE 4.x is only used to provision Windows 8 and Windows 2012.

Red Hat Before capturing a 64 bit Red Hat Linux machine to be migrated, perform the following steps:

1. Backup the existing initrd file located in the `/boot` directory.
2. Modify `/etc/modprobe.conf` by adding these lines (if additional `scsi_hostadapter` entries are already in the file, increment only the counter):
   ```
   alias scsi_hostadapter1 mptbase
   alias scsi_hostadapter2 mptspi
   alias scsi_hostadapter3 ata_piix
   alias scsi_hostadapter<N> <module>
   ```
3. Run: `mkinitrd /boot/initrd-<kernelversion>.img <kernelversion>`

Note: Before deploying an image captured from a different source, for example, a physical computer, on a VMWare ESX hypervisor, ensure that you created the new virtual machine to be deployed with type Microsoft Vista, even if another operating system is going to be deployed on that virtual machine. It is recommended that you use a BUS LOGIC disk adapter and a RAM of at least 1 GB.

You need to inject the appropriate BUS LOGIC mass storage drivers in the WinPE software module.

It might be necessary to inject also the network adapter drivers, but if your virtual machine has the e1000 network adapter model, you do not need to inject its related drivers.

If you capture an image that has four primary partitions, this image cannot be deployed. The fourth partition must be a logical partition. You cannot also deploy an image in which the OS partition follows the data partition.

The product cannot clone, capture an image, or perform direct migration from a target with unformatted partitions, or partitions formatted using a proprietary file system that it does not support. Such partitions should be either deleted, or formatted using a supported file system before cloning, capturing an image, or migrating to another computer.

You can capture an image online that is, by running on the source machine a process that selectively reads all the relevant data from the hard disks, and saves it optimally in a virtual image, defragmenting all files ad-hoc.

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **Advanced features** > **Image Monitor**.
3. Select **Add a new image** from the contextual menu and click **Next**.
4. Select the source machine that you want to capture. It can be either a physical or a virtual machine.

   Note: Ensure that the source machine

   - Is currently powered on
   - Is connected to the server, that is the machine was booted using PXE or a network boot media
5. Enter an image description.
6. Click **Next** to start the process to create the virtual image.

The image of the computer, hypervisor, or virtual machine that you selected has been captured. The capture process creates an OVF file and some disk files in the OS deployment server format. The creation of the virtual image might be slower than the standard profile creation.

You can now see the details of the captured image, by clicking **Server** > **Advanced features** > **Image Monitor**.

You can also deploy this image by clicking **Server** > **OS deployment** > **Target Monitor** and then **Deploy now** > **A virtual image** from the contextual menu, or migrate it to another repository by clicking **Server** > **Advanced features** > **Image Monitor** and then **Replicate to another server** or **Export to OVF** from the contextual menu.

Types of virtual image captures

You can capture an image by using an offline or an online method.

The standard process of capturing and restoring an image is the offline method, which captures the image without analyzing the filesystem.

The actions that are based on an offline capture and restore algorithm are:

- Replicating an image to another OS deployment server. See “Replicating an image to another OS deployment server” on page 322.
- Exporting an image to the OVF format. See “Exporting images to OVF format” on page 323.
- Importing an image from the OVF format. See “Importing images from OVF format” on page 325.

The offline method of capturing images is very simple and fast, but it has the following disadvantages:

- Consumes a lot of space because it does not identify the sectors that are no longer used
- Captures a lot of temporary files, with security exposures
- Does not defragment virtual disk and filesystem
- Does not work with physical machines

The actions that are based on an online capture and restore algorithm are:
- Capturing images and storing on the server
- Deploying a virtual image
- Deploying an operating system from an existing machine (direct migration)

Note: The online operations are not supported in the XEN environment.

The online image capture is more complex and can be slower than the offline capture, because it reboots the Tivoli Provisioning Manager for Images environment, but it has the following advantages:
- Implicit defragmentations of the virtual disk and filesystem
- Takes smaller footprint of the virtual disk
- Is more secure
- Provides a faster restore and deployment process
- Works on virtual and physical machines

Browsing images

You can display all images in your environment by using the Image Monitor feature.

1. To see all the images in your environment, log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **Advanced features** > **Image Monitor**. Each image is identified by a name and a status. Possible statuses are:

   snapshot images
   It identifies images discovered on your hypervisors or created on your Tivoli Provisioning Manager for Images server.

   current
   It identifies the current status of virtual machines discovered on your hypervisors.
3. To see additional details about a specific image, select it and click **View image details** from the contextual menu.

You now know all the details of snapshot and running images, such as the image hierarchy that describes the image origins, the image status, the image type, the image content, and the different image locations.

Note: Image release information is provided only for snapshot images which belong to an image hierarchy. No value is provided for current images or for snapshot images without of an image hierarchy.

From the **Image Monitor** page you can now export these images to OVF format or migrate them to another repository. You can also deploy them from the **Target Monitor** page.

Binding drivers to a Windows image

When an image does not contain the drivers needed for deployment, you must bind these drivers to the image to be able to deploy it and obtain a working operating system.

If you encounter problems with the built-in drivers contained in your image, if some drivers are not bound, or if some drivers are missing, you should bind other drivers to your image.

You can only bind drivers to your image that are software modules in your OS deployment server. You must therefore create driver software modules from the drivers that you want to bind to your image.

Note: There are two methods to bind driver software modules to an image:

- the *standard binding rule method* where you can indicate images to bind to a software module.
- the *driver-specific binding rule method* where you bind drivers per image and target model/device pair.

You can switch from one method to the other. In the *driver-specific binding rule method*, driver bindings from the *standard binding rule method* are ignored, and vice-versa.

The method described here is the *driver-specific binding rule method*.

From version 7.1.1.3 of the product onwards, it is recommended to use the *driver-specific binding rule method* whenever possible, which is the method by default on all newly captured Windows images. The *driver-specific binding rule method* is available for images captured with the wizard and for images imported from OVF.

The product helps you select appropriate drivers for particular target models. It helps you to predict potential problems and to solve them. It does not guaranty that a specific image, with bound drivers, works with a given target.

The information used by the OS deployment server to predict the compatibility of a driver with a target model is taken from the content provided by the vendor in its driver. The OS deployment server cannot verify the accuracy of this information.

1. Check the compatibility of your image.
   a. Go to **Server** > **Advanced features** > **Image Monitor**.
   b. To view the details of the image, you have two options.
      - Double-click it.
      - Select an image, and then select **View image details** in the contextual menu.
   c. *(Optional)* In the section **Driver handling**, click **Switch to driver specific bindings mode**. You only need to perform this step if you are in the regular software binding rule mode.

d. A check is performed while the page is loading. This may take a few minutes. By default, checks are performed on all available drivers.

   If drivers are missing, or are not bound, or if several drivers are bound for the same device, the following information is provided:

   ❌ Indicates a missing critical driver, or a critical driver of the wrong architecture.

   ✖ Indicates that a missing non-critical driver, or a non-critical driver of the wrong architecture.

   ⚠ Indicates that a required driver is present on the OS deployment server, but that it is not bound.

   ℹ Indicates that there are several drivers bound for the same device, or that there is a binding with a driver that is not known as compatible.

   You can expand the item to get more information.

   - For drivers missing on the OS deployment server, you can discover where to look for it, including, if available, a download link and the exact directory within the downloaded archive where the driver can be found.
   - When drivers are present on the OS deployment server, you can discover which driver to bind, in order of preference. If multiple drivers are known to possibly work for a device, the best choice is listed first. The choice is explained in the advice text, which first recommends the use of *device-specific drivers*, that is, drivers that have been specifically designed for the given hardware device. Then *compatible device drivers*, that match the device family, are recommended, even if they are not an exact rebranded variant (for example, as second choice, an Adaptec driver of the same family as an IBM ServerRaid adapter, if it is based on the same chipset). Finally, as third choice, *generic drivers*, for example, Microsoft generic AHCI driver for any AHCI controller, are recommended.

   If no error is found, you do not need to modify the bindings.

2. Modify the driver bindings of the image. There are two ways to perform this.
   - Use a wizard.
     a. Click **Fix Drivers**.
     b. Follow the instructions of the wizard. After having selected a target model, you have to select one of these options:

        Automatically fix issues which can be fixed for this model.
        Fixes all issues which can be automatically fixed. Such issues include a missing binding to an existing driver, or multiple bindings for a device, for example.

        Manually fix issues for this model.
        Presents you with each issue in turn. Ways to solve the issue, when available, are proposed.

        Automatically bind drivers for this model.
        Erases every existing binding. New bindings are then automatically added.

        Copy driver bindings for this model from a similar image.
        Copies all the bindings from a selected source image to the current one.

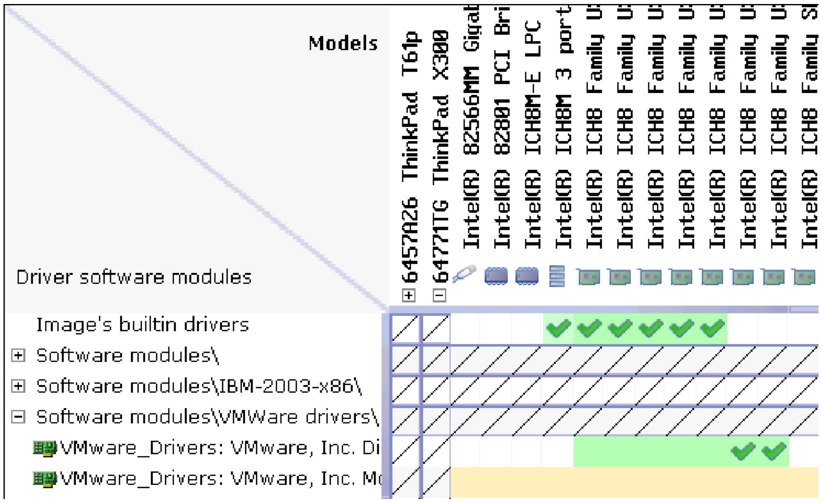
        Reset all drivers bindings for this model.
        Erases all the driver bindings, and does not create any new binding.

- Edit the bindings manually.
  a. Click **Edit image's driver bindings** on the **Image details** page.

     A grid is loaded.

     Columns represent target models known to the OS deployment server. They can be expanded to view their devices, provided an inventory has been performed.

     The first line represents the image. Other lines represent software module folders in the OS deployment server. They can be expanded to view individual drivers. If a driver can be used only for 32-bit or 64-bit machines, a superscript *x86* or *x86-64* mark is written next to the driver name. If you do not find the drivers that you need in the list provided, you should first create software modules for your drivers.
  b. Optional. To obtain a summary of the errors and warnings, click the link above the grid. This helps you locate the problematic areas in the driver grid.
  c. Expand the columns of problematic target models to view individual devices.
  d. Expand software module folders containing drivers to view the individual drivers.



Driver software modules \ Models	6457A26 ThinkPad T61p	64771TG ThinkPad X300	Intel(R) 82566MM Gigat	Intel(R) 82801 PCI Bri	Intel(R) ICH8M-E LPC	Intel(R) ICH8M 3 port	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family U	Intel(R) ICH8 Family Sl
Image's builtin drivers						✓	✓	✓	✓	✓	✓			
⊞ Software modules\														
⊞ Software modules\IBM-2003-x86\														
⊟ Software modules\VMWare drivers\														
VMware_Drivers: VMware, Inc. Di												✓	✓	
VMware_Drivers: VMware, Inc. Mc														

Figure 4. Driver binding grid

A cell with a green background indicates that driver information corresponds to the device. The quality of the drivers that can be selected is illustrated by the intensity of the green background: ■ ■ ■ the best drivers are in intense green, the family drivers are in standard green, and the generic drivers are in pale green.

A cell with an orange background indicates either that the driver is not a PCI driver, or that there is no compatibility information available for the driver.

A cell with a green check mark ✓ indicates that the driver is bound to the image for use with the specific target model and device.
  e. Click on a green background cell to add or remove bindings.

     It is not possible to bind or unbind drivers from the image itself, because they are built in drivers.

You should have one, and only one, check mark per column, indicating that you have one and only one driver for each device.

f. When you are done modifying the bindings, click **Save**.

g. To return to the **Image details** page, click **Back**.

Potential problems with the image are recomputed, allowing you to check if your modifications have solved the detected problems.

When you have solved all the driver binding issues, you can deploy targets with your image.

Deploying virtual images and machines

You can deploy from a physical machine to a virtual or physical machine, and from a virtual machine to a virtual or physical machine. For example, you can deploy from a physical server to an ESX guest, from a physical laptop to a KVM guest, from a Hyper-V guest to a physical machine, or from an ESx guest to a KVM guest. In particular you can deploy any virtual image or machine defined in Tivoli Provisioning Manager for Images on target that can be computers, hypervisors, physical servers, or virtual machines.

Note: When an image is deployed to a destination target which has a significantly different hardware from the source target of the image, you may encounter driver and/or disk mapping issues. This is particularly true for Linux operating systems where driver binding is not available, as well as for 64-bit operating systems.

The same issues may also arise for direct migration when the source hardware and destination hardware are significantly different.

Note: Before you deploy a virtual image, ensure that both the destination target and the source target have the same amount of disks.

Partition layout in virtual image deployment and machine migration

When deploying a virtual image or migrating a machine, the original deployment layout may be altered on the target.

- Primary partitions are created in priority. If more than three partitions must be created, the remaining ones are logical partitions.
- The largest partition is always expanded to fill all the disk space left available by the other partitions.

Deploying virtual images

You can deploy any image defined in Tivoli Provisioning Manager for Images on targets that can be computers, hypervisors, or virtual machines. The Image Monitor has this list of available images to be deployed.

Attention: You cannot deploy the following images:

- Images that have 4 primary partitions. The fourth partition must be a extended partition.
- Images in which the OS partition follows the data partition.

Windows If you deploy a Windows virtual machine, ensure that you created a WinPE deployment engine.

Vista 2008 Windows 7 Windows 8 2012

- To deploy a Windows Vista/2008/7/2012/8 32-bit virtual image, the minimal size of the hard drive of your target is about 10 GB.
- To deploy a Windows Vista/2008/7/2012/8 64-bit virtual image, the minimal size of the hard drive of your target is about 20 GB.

Vista 2008 Windows 7 For provisioning Windows Vista, Windows 7, and Windows 2008 operating systems, you must have stored a WinPE 3.x deployment engine on the OS deployment server.

2012 Windows 8 For provisioning Windows 8 and Windows Server 2012 operating systems, you must have stored a WinPE 4.x deployment engine on the OS deployment server.

2012 For Windows 2012, since the password is required and cannot be empty, you must set the password in the target details for the migration to complete fully unattended otherwise the task will stop on the target side asking to set a password for the Administrator and you must manually insert it to let the task continue.

Linux To deploy your virtual image with a type of hard disk different from that of your reference target (for example, going from a parallel hard disk to an SCSI or an AHCI disk), the system handles hardware changes by rebuilding the initial ramdisk (or initrd) during deployment. The rebuilding of initrd is available for 32-bit cloned images only.

Vista 2008 Windows 7 The user name and administrator password are set to the values specified in the target details for each target. The user name needs to be set for a successful deployment. If the password value is not specified, the default value of the administrator password is `""`.

2012 Windows 8 As the password is required and cannot be empty, you must set the password in the target details for the migration to complete fully unattended, otherwise the task will stop on the target side asking to set a password for the Administrator and you must manually insert it to continue the task.

To deploy a virtual image means to deploy an image captured on the OS deployment server or discovered on an hypervisor and displayed by the Image Monitor as an image having the status `Snapshot`.

To deploy an image on multiple targets, perform the following steps:

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **OS deployment** > **Target Monitor**.
3. Select a single target or multiple targets on the Target Monitor page. To select multiple targets, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the **Ctrl** key.
4. Click **Deploy now** from the contextual menu.
5. Click **A virtual image (system snapshot)**. If you want to install additional software, you can also click **Install additional software**.
6. Choose to deploy or redeploy your targets.
7. Select a deployment scheme to define how the OS configuration is deployed.

8. Select the virtual image to deploy.
9. If you clicked also the option **Install additional software**, select the software modules that you want to bind to and deploy on the selected targets.
10. To deploy the image on the targets without physically interacting with the computers, you can start up or reboot them automatically by selecting one of the following options:
    - **Try to wake up targets currently powered off using WOL** asks the Target Monitor to send IBM Wake on LAN packets to wake up targets. Waking up targets only works on carefully designed modern computers. A target can only be woken up if its network adapter and its system board support Wake on LAN packets, and if the network adapter has been shut down correctly. If the network adapter is not in the appropriate power state, Wake on LAN packets cannot wake the computer up. This is not specific to Tivoli Provisioning Manager for Images, but is a general limitation of the Wake on LAN technology.
    - **Try to wake up targets using management interface** tries to turn on the target using command lines to the management interface. This works only for targets with an interface management known to the product and properly identified.
    - **Try to reboot targets running web interface extension** asks the Target Monitor to contact the targets if they are running under Windows and send a reboot request. If you are not running Windows, you cannot reboot targets remotely, but must reboot them manually. If you are running Windows, you need the web interface extension running with the correct privileges to run a remote boot.

    If you have not selected any of these options, or if they do not work, and if the targets on which you are trying to deploy are not powered on, turn them on now and make them start on the network.
11. You can set the schedule for the deployment and then click **Next**.

The image has been deployed on the selected targets. Because during deployment the largest partition of each disk is extended to fit the destination disk size, the source partition scheme might not match with the destination one.

You can now use the software deployed on the targets.

Migrating machines

Using Tivoli Provisioning Manager for Images you can deploy a machine, which can be a computer, a hypervisor, or a virtual machine directly on targets, which can be computers, hypervisors, or virtual machines.

Ensure that both the source and the target machines boot from the network or from the netboot CD.

The product cannot clone, capture an image, or perform direct migration from a target with unformatted partitions, or partitions formatted using a proprietary file system that it does not support. Such partitions should be either deleted, or formatted using a supported file system before cloning, capturing an image, or migrating to another computer. Vista 2008 Windows 7

- To migrate a Windows Vista/2008/7/2012 32-bit operating system, the minimal size of the hard drive of your target is about 10 GB.
- To migrate a Windows Vista/2008/7/2012 64-bit operating system, the minimal size of the hard drive of your target is about 20 GB.

Vista/2008/7/2012

For Windows 2012, since the password is required and cannot be empty, you must set the password in the target details for the migration to complete fully unattended otherwise the task will stop on the target side asking to set a password for the Administrator and you must manually insert it to let the task continue.

Linux To deploy your virtual image with a type of hard disk different from that of your reference target (for example, going from a parallel hard disk to an SCSI or an AHCI disk), the system handles hardware changes by rebuilding the initial ramdisk (or initrd) during deployment. The rebuilding of initrd is available for 32-bit cloned images only.

Linux To deploy your virtual image with a type of hard disk different from that of your reference target (for example, going from a parallel hard disk to an SCSI or an AHCI disk), the system handles hardware changes by rebuilding the initial ramdisk (or initrd) during deployment. The rebuilding of initrd is available for 32-bit cloned images only.

Vista 2008 Windows 7 The user name and administrator password are set to the values specified in the target details for each target. The user name needs to be set for a successful deployment. If the password value is not specified, the default value of the administrator password is "".

2012 Windows 8 As the password is required and cannot be empty, you must set the password in the target details for the migration to complete fully unattended, otherwise the task will stop on the target side asking to set a password for the Administrator and you must manually insert it to continue the task.

To perform a direct migration means to migrate a running machine displayed by the Target monitor on the OS deployment server, and perform a direct operation from source to target without using the OS deployment server.

To migrate a machine, perform the following steps:

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **OS deployment** > **Target Monitor**.
3. Select a single target or multiple targets on the Target Monitor page. To select multiple targets, select an administrative group, a custom list, a subnet, or click on individual target names while holding down the **Ctrl** key.
4. Click **Deploy now** from the contextual menu.
5. Select the option **An existing machine (direct migration)**. If you want to install additional software, you can also click **Install additional software**.
6. Choose to deploy or redeploy your targets.
7. Select a deployment scheme to define how the OS configuration is deployed.
8. Select the virtual machine to migrate.
9. If you clicked also the option **Install additional software**, select the software modules that you want to bind to and deploy on the selected targets.
10. To migrate the machine on the targets without physically interacting with the computers, you can start up or reboot them automatically by selecting one of the following options:

- **Try to wake up targets currently powered off using WOL** asks the Target Monitor to send IBM Wake on LAN packets to wake up targets. Waking up targets only works on carefully designed modern computers. A target can only be woken up if its network adapter and its system board support Wake on LAN packets, and if the network adapter has been shut down properly. If the network adapter is not in the appropriate power state, Wake on LAN packets cannot wake the computer up. This is not specific to Tivoli Provisioning Manager for Images, but is a general limitation of the Wake on LAN technology.
- **Try to wake up targets using management interface** tries to turn on the target using command lines to the management interface. This works only for targets with an interface management known to the product and properly identified.
- **Try to reboot targets running web interface extension** asks the Target Monitor to contact the targets if they are running under Windows and send a reboot request. If you are not running Windows, you cannot reboot targets remotely, but must reboot them manually. If you are running Windows, you need the web interface extension running with the correct privileges to run a remote boot.

If you have not selected any of these options, or if they do not work, and if the targets on which you are trying to deploy are not powered on, turn them on now and make them start on the network.

11. You can set the schedule for the migration and then click **Next**.

The machine has been migrated on the selected targets.

Note: With direct migration, some hardware-specific components might not get automatically configured correctly. This happens in particular for the X server, because the X accelerated engine might need to be changed, which requires manual steps.

You can now use the migrated machine on the selected targets.

Replicating an image to another OS deployment server

Using Tivoli Provisioning Manager for Images you can replicate an image located on an hypervisor to a OS deployment server or an image located on a OS deployment server to another OS deployment server. For example, you can replicate an image already captured on an hypervisor to a OS deployment server, or an existing image can be replicated from a OS deployment server to another OS deployment server.

If you replicate an image located on an hypervisor, install the rbagent, also known as web interface extension, on the hypervisor and start it using the `rad-refreshhypervisor` command as in the following example:

```
rbagent.exe -s host_ipaddress:host_password rad-refreshhypervisor
```

Ensure that there is enough disk space on the OS deployment server where you want to replicate the image. Turn on the OS deployment server target before starting the replication.

Note:

- With a multiple server architecture, you can replicate system profiles, software modules, deployment schemes, and hardware configurations from a parent server to its child.
- You can replicate a virtual image between servers only if the servers share the same database.
- The driver binding grid is applicable when performing a direct migration (replication).

All the images displayed in the **Image Monitor** page are located on the OS deployment server or have been discovered in a hypervisor. You can replicate the images that are defined in the OS deployment server database, to any OS deployment server.

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **Advanced features** > **Image Monitor**.
3. Select the image that you want to migrate.
4. From the contextual menu, select **Replicate to another server**.
5. Specify the IP address of the OS deployment server where you want to copy the selected image.
6. Click **Next** to start the process to copy the image to the specified OS deployment server.

The image has been replicated to the specified OS deployment server.

Note: When replicating a virtual image with status `Snapshot`, the related image entry is updated in the Image Monitor by adding the new OS deployment server location for this image. The same image has a new location.

When replicating a virtual image with status `Current`, an image is created and a new entry name ending with the `'Replicated'` string is displayed in the Image Monitor. This entry shows that you have replicated a running image. The replicated image has the same parent image as the source one.

You can now see the new location of the image, by clicking **Server** > **Advanced features** > **Image Monitor**.

Exporting images to OVF format

Using Tivoli Provisioning Manager for Images, you can export existing images to an OVF file.

- Install the rbagent, also known as the web interface extension, in the background as a service. If you want to export in the VMDK format, you must install the web interface extension in its 32 bit version to be compatible with VDDK.
- Ensure that there is enough disk space on the machine on which you want to export the image.

  Note: To export an image to OVF format on Windows systems, ensure that the image is less than:
  - 2 GB, on a FAT16 filesystem
  - 4 GB, on a FAT32 filesystem

  On an NTFS filesystem, the only image size limit is the available hard disk space. The exFAT filesystem is not supported.

To export an image to the OVF format, perform the following steps:

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **Advanced features** > **Image Monitor**.
3. Select the image that you want to export.
4. Click **Export to OVF** from the contextual menu.
5. Select the computer where you want to export the OVF image. You can select the local computer if web interface extension is installed, the OS deployment server, or any computer with the web interface extension installed.
6. Specify the path where you want to export your OVF image. If you choose the OS deployment server, the path is the local `<TPMfOSD_DATA_DIR>\import` directory. A virtual image in OVF format consists of a file with the .ovf extension, and a set of virtual disk image files.
7. Select the format to use for the virtual disk image files, depending on the format supported by the tool with which you will reimport the image later. Available formats are: VHD, VMDK, and raw sparse.

   Note: If the selected virtual disk image format is VMDK, before exporting the image ensure that:

   - You installed the VDDK library on the machine where the VMDK file will be located.

   When installing VDDK, change the environment variables as follows:

   - On Windows systems, add the binary path (such as `C:\Program Files\VMware\VMware Virtual Disk Development Kit\bin`) to the `%PATH%` variable. If you are using a 64-bit rbagent or the import directory of a 64-bit server, add to the `%PATH%` variable the bin directory by unzipping the `vddk64.zip` file contained in the VDDK bin directory (such as `C:\Program Files\VMware\VMware Virtual Disk Development Kit\bin\vddk64.zip`). Refresh the environment variables. If needed, restart your machine.
   - On Linux systems, add the library installation path (such as `/usr/lib/vmware-vix-disklib/lib32`) to the `/etc/ld.so.conf` file. If you are using a 64-bit rbagent or the import directory of a 64-bit server, add the 64-bit library installation path (such as `/usr/lib/vmware-vix-disklib/lib64`) to the `/etc/ld.so.conf` file. Refresh the settings by running **ldconfig**. Refresh the environment variables. If needed, restart your machine.

   Before using the exported VMDK file on a VMWare ESX Server, you must run the following command on the hypervisor:

   ```
   vmkfstools -i <source_bytpmfosd> <output_for_esx>
   ```

   where:

   <source_bytpmfosd>
   Specifies the name of the VMDK file created by using Tivoli Provisioning Manager for Images.

   <output_for_esx>
   Specifies the name of the destination VMDK file to run on the VMWare ESX Server.

   For example:

   ```
   vmkfstools -i ./OVFSLES10_OM.vmdk
   ./SLES10_CONVERTED/OVFSLES10_OM_CONVERTED.vmdk
   ```

Using the generated `<output_for_esx>` file you can run the new virtual machine on the VMWare ESX Server.

The image has been exported on the specified computer in the OVF format.

You can now use the new image on the targets.

Note: If the images are exported to OVF in a format different from the format of the source hypervisor, they might be unusable on the target hypervisor. This is because of the different hardware emulation. To avoid this problem, use the standard capture and deployment process on virtual machines, because this process rebuilds the system according to the different hardware.

Importing images from OVF format

Using Tivoli Provisioning Manager for Images you can import an existing virtual image into your image repository stored on the server. You must provide the virtual image in OVF format, together with all virtual disk images referenced by the OVF file.

For more information about the OVF standard, see the Open Virtualization Format Specification available from the DMTF Web site at: http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.0.0.pdf

- Install the rbagent, also known as web interface extension, in the background as a service on the machine where the virtual image is located. The web interface extension is necessary to import files remotely from any computer. If you do not want to install the Web interface extension, make your files available in the OS deployment server `<TPMfOSD_DATA_DIR>\import` directory so that the OS deployment server can access them directly.
- Ensure that there is enough disk space on the OS deployment server `<TPMfOSD_DATA_DIR>\vimages` directory where the image is imported.
- If the virtual disk images referenced by the OVF file have a VMDK format, before importing the image ensure that:
  - You installed the VDDK library on the machine where the VMDK file is located. A reboot is required after VDDK installation.
  - You defined the VDDK path in the library path of the machine root user.

To import an image from the OVF format, perform the following steps:

1. Log on to the Tivoli Provisioning Manager for Images interface.
2. Click **Server** > **Advanced features** > **Image Monitor**.
3. Click **Import from OVF** from the contextual menu.
4. Select the computer where the OVF image is located. It can be the local computer, the OS deployment server (in the `<TPMfOSD_DATA_DIR>\import` directory), or any computer with the web interface extension installed.
5. Specify the full path of the OVF image.

The image has been imported from the specified computer in the OVF format.

You can now deploy the new image on the targets.

Discovering virtual machines and virtual images

Using Tivoli Provisioning Manager for Images you can discover all the virtual machines and virtual images of a hypervisor. The virtual machines that are discovered can be seen in the **by Hypervisor** custom list.

- Install the rbagent, also known as web interface extension, on the hypervisor with administrator privileges. For more details, see “Hypervisor requirements” on page 304.
- On VMWare, to correctly mount the VMDK files, ensure that the virtual machine is powered off.
- To correctly discover VMWare environments, on the OS deployment server ensure that:
  - The VDDK library is installed
  - The VDDK path is defined in the environment variables of the OS deployment server.
  - After running the VDDK library installation, the OS deployment server has been rebooted.

Note: The virtual image discovery performed by the web interface extension installed on the hypervisor also performs a PCI inventory of each discovered guest. This PCI inventory discovers information related to NET and SCSI devices only. To run a complete PCI inventory perform a PXE or USB/CD boot of the virtual image.

Note: For virtual machines running the Linux operating system, you can only discover the content of computers with ext2 or ext3 file systems of 128-byte inode size. To determine if the file system has a 128-byte inode size, run the following command:

```
tune2fs -l /dev/sda2 | grep "Inode size"
```

where `/dev/sda2` is the file system where the operating system is installed.

Note: For virtual machines running Red Hat Enterprise Linux (RHEL) 6 operating systems, the Image Monitor panel does not display the OS architecture (x86 or x86-64) .

You can discover virtual machines and virtual images located on VMWare and KVM (with or without libvirt library) hosting environments.

1. To discover all the virtual machines and virtual images of a hypervisor, log on to the Tivoli Provisioning Manager for Images interface.
2. Go to **Server** > **OS deployment** > **Target Monitor**. In the **by Hypervisor** folder you can find all the discovered hypervisors.
3. Select a hypervisor target on which to run a complete discovery.
4. In the contextual menu, select **Additional features**.
5. Select **Run a host inventory**.
6. Click **Next** and follow the instructions of the wizard. All the information about the operating system, software modules, and virtual images is discovered and stored on the Tivoli Provisioning Manager for Images database.

   Note: If you installed the rbagent as a command-line tool, to run a complete discovery from the command line, enter the following command on the hypervisor:

   ```
   rbagent.exe -s host_ipaddress:host_password rad-hostinventory
   ```

where:

host_ip_address
The IP address of the OS deployment server.

host_password
The password of the OS deployment server. It is not mandatory to set the IP address if the rbagent.conf already exists. You can create this file from the Web console or when installing the web interface extension.

This command checks for any images or virtual machines that have been created or deleted. It does not discover the existing virtual machines to see, for example, if a software has been added or removed.

To run an inventory every minute, enter the following command:

```
rbagent.exe -s host_ipaddress:host_password rad-refreshhypervisor
```

The virtual images and virtual machines of the target have been completely discovered.

You can now see the details, by clicking **Server** > **Advanced features** > **Image Monitor**. You can also deploy the image to a physical or virtual machine or copy these images to another repository.

Chapter 8. Multiple server architecture

A key with parent and child servers is keeping them replicated in order for individual targets to be deployed with the appropriate content.

When building a multiple server architecture, there are two main design choices

- Several OS deployment servers connected to a single, centralized database.
- Several OS deployment servers connected to several databases.

Setting up the architecture and then replicating the information from one server to another depends on these options. More details on the two different architectures and how to set them can be found in the Tivoli Provisioning Manager for Images Installation Guide, in appendix A.

It is important to remember the following points when performing replication online:

- Each subordinate server needs to download files from its parent server. This means that the parent server must be up and running during the whole replication process.
- Replication can be scheduled for a specific time and repeated at an interval specified in days. These settings are set on the subordinate server. When set, the replication process becomes autonomous and can be performed without human intervention. Drawbacks to this are the relative loss of control over the process, and network and processor usage.
- Server replication is performed by copying files from the parent server to the subordinate server. A selection can be performed on the kind of information that must be replicated. Files that have been modified are copied over.

Note: If you use the `config.csv` configuration file to setup your server architecture and its options, do not use the links provided on the **Server** > **Server parameters** > **Server replication** page to create your architecture. Contradicting information provided on the web interface and through the `config.csv` file leads to unexpected behavior.

Note: If you work with a multiple server architecture, you must be aware that the `\TPMfOS Files\tftp` folder is not replicated from parent to child servers. Therefore, if a task is triggered on the parent server but the target boots on the child, you must use an external replication mechanism to ensure that the parent and child versions of this folder are synchronized.

Server roles

In a multiserver architecture, roles are associated with OS deployment servers.

The server role of an OS deployment server in a multiserver hierarchy depends on whether the server has its own database and whether targets are replicated.

Server role information is displayed in **Server** > **Server parameter** > **Server replication** when a server is selected in the hierarchy.

Single database architecture and multiple database architecture must be considered separately. In the diagrams, servers are grouped according to the database they use.

Single database architecture

In a single database architecture, there are only two possible server roles as exemplified in Figure 5:

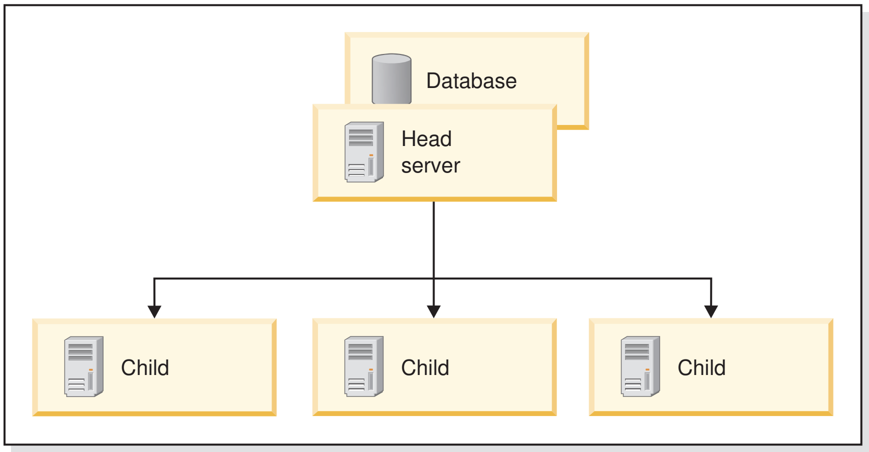


Figure 5. Single database architecture

Head server
The server at the top of the hierarchy.

Child All other servers.

Multiple database architecture

The structure of a multiple database architecture is more complex and more roles can enter into play, as exemplified in Figure 6 on page 331.

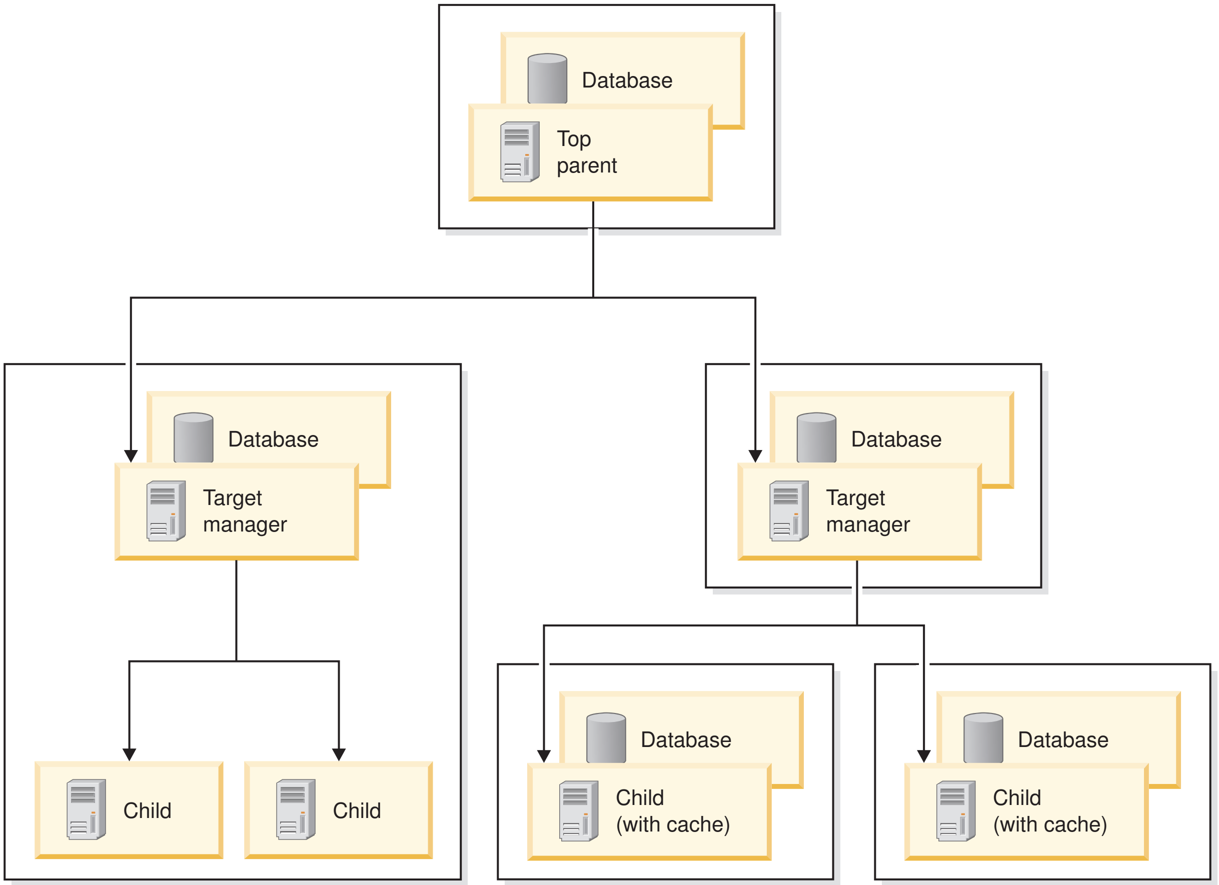


Figure 6. Multiple database architecture

Top parent
The server at the top of the hierarchy. It manages its database and does not have a parent database.

Target manager
A replicated server, with its own database, and which manages its targets, that is, it does not report target status to a parent server.

Child A server without its own database, as in the single database architecture.

Child (with cache)
A server with its own database, with full replication, and at the bottom of the hierarchy.

OS deployment server replication

Replication is the means to keep databases, files, and information up-to-date from parent to child servers. It can be performed through several mechanisms.

Replicated objects

The objects that are replicated from a parent to a child OS deployment server are the following:

- Deployment schemes
- Hardware configurations
- Software modules
- System profiles

Targets, virtual images, and server tasks are not normally replicated.

Main steps in server replication

Database replication for multiple database architecture only

A verification is performed on a regular basis between parent and child databases. Any discrepancy in the databases indicates that the files on the child server are not up-to-date. The child database must be updated. The default interval between two verifications is one minute, by default, but this can be configured in the `config.csv` file.

File checking against the database

A verification is performed between the files on the OS deployment server and the database. The web interface reflects any need of file replication.

File replication

Once the database verification has uncovered file discrepancies, the new and updated files must be downloaded from the parent server to the child server.

Server replication techniques

There are several ways to perform server replication depending on whether you use a single database or multiple databases, the type of network connectivity between the servers and the databases, and if you keep a strict top-down hierarchy between your servers.

Online, with a single database

If you have a single database and a good network connection between your servers and your database, you can opt for one of the following replication mechanisms:

- Automatically each time needed, with a `config.csv` file
- Automatically at scheduled times
- Manually using the web interface or a command-line (web interface extension).

Online, with multiple databases

If you have multiple databases and good network connectivity between your servers, you can set up replication through the `config.csv` file.

Offline, with the web interface extension

If you cannot ensure that your servers are always connected, you can replicate with the web interface extension and a specific package called, `sync.pak`. This method assumes a strict top-down hierarchy between your servers. All the deployment objects located on the parent server are replicated down to the children.

Offline, with RAD files

If you cannot ensure that your servers are always connected, but want to replicate only some objects from one server to another, you can export RAD files from one server and import them onto another.

Online, one time replication

In some cases, you can perform a one-time replication between servers, for example, if you want to change the database, or if you want to make a copy of a production server to perform tests on the copy. This requires reliable network connectivity.

Online, with the Java API

To replicate specific objects from one OS deployment server to another, you can use the Java API. See the documentation on using the Java API in the product in the sectionJava API.

Multiple server architecture considerations

If you delete an object from a child server, the parent server is not aware of the deletion. When you replicate, the object is recreated on the child server.

Replication is performed top-down, from parent servers to child servers.

Tasks must be started at the appropriate level in the server architecture. A task cannot normally be initiated on a parent to be performed on a child server.

To keep production servers in a clean state, it is recommended to create and test all replicated objects on a dedicated test server. When an object is ready for production, export it to a RAD archive and import it at the right level of the production hierarchy.

Multiple database architecture considerations

When an object is created, modified, or deleted on a parent server, one *PollInterval* should elapse to allow for the propagation of the changes to the database before replication is triggered. If the database is updated during the replication, the replication tasks fails.

When a child server is stopped for longer than three times the *PollInterval* (*PollInterval* is 1 minute by default), it performs a full refresh, checking the database, all objects, and all files when restarting.

The value of *PollInterval* can be updated in `config.csv`.

If you delete an object from a child server, the parent server is not aware of the deletion. When you replicate, the object is recreated on the child server.

Targets known to a child server are unknown from its parent because target information is stored in separate databases and is normally not replicated. This implies that you cannot capture an image or clone a system profile from a parent server if the reference target is known only by the child server.

Building a hierarchy between two or more servers with heterogeneous databases

When you build a hierarchy between two or more servers with heterogeneous databases, such as a Windows parent with a DB2 database and a Linux or UNIX child with an Apache Derby database, ensure that the database replication is performed using the JDBC gateway instead of the ODBC gateway. Use the ODBC gateway only if all servers run on Windows.

Moreover, all databases must use the same collation.

Replicating virtual images

Virtual images are not part of the set of objects which are replicated from a parent to a child OS deployment server. To replicate a virtual image from one server to another, you must do so explicitly for every image you want to replicate.

Exceptions to the replication flow and to the replicated objects

Under specific circumstances, it is possible to replicate some target information from parent to child servers, and from child to parent servers. In the same circumstances, it is also possible to replicate a deployment task. For this, you must do the following:

- Set up replication for a **child (with cache)** server. This server is at the bottom of a multiple database architecture, it has its own database which it does not share with another server. For details about the **child (with cache)** server, see “Server roles” on page 329.
- Use a text file (`config.csv`) to configure the replication.
- Set flag `h` in **`AutoSync`**, in the configuration text file.

In this particular configuration, target information is replicated top-down and bottom-up. A target added to a child server can therefore be known to its parent server. Moreover, a deployment task can be started on a parent server to be run by a child.

In a multi-server parent-child hierarchy with multiple databases and with the target replication configured (flag "h" in the `config.csv` file) before Fix Pack 12 the target replication (that is bidirectional, both from the child database to the master database and viceversa) did not include the replication of hardware inventory information. As a result, if a new hardware model appears on a child server, the target displays on its parent but, due to the missing inventory information, it was not possible to update the deployment engines to include drivers for the new hardware model. Starting from Fix Pack 12, when a target performs a PXE boot and logs into an OS Deployment server, the server while collecting the inventory information will propagate this information. The replication of the hardware inventory information will be unidirectional (upwards only).

Important:

In a Multiserver hierarchy, the database names must be different, even if on different database servers.

Replicating OS deployment servers with a schedule

To replicate your OS deployment servers regularly, you can set up a replication schedule, indicating the frequency of the replication

If you have a hierarchy of more than two levels of parent and child servers, the scheduling must match the hierarchy. Top servers must be replicated first, and child servers after.

- With a multiple database infrastructure, edit the `config.csv` file to include the new schedule.
- With a single database infrastructure, use the web interface to set up a replication schedule. For each child server in the hierarchy
  1. Go to **Server** > **Server parameters** > **Server replication**.
  2. Click **Set up a replication schedule**.
  3. Enter the start date and time, and the frequency of the replications. As child servers must be replicated after their parents, you must use the same or a lower frequency than the replication schedule on the parent server.

4. Click **OK**.

Replicating an OS deployment server once manually

Replicating OS deployment servers can be done manually with the web interface or with the web interface extension.

- With the web interface:
  1. Go to **Server** > **Server parameters** > **Server replication**.
  2. Select the OS deployment server you want to replicate.
  3. Click on the link to replicate the server. The exact wording of the link depends on whether the server needs to replicate, and on the position of the server in the hierarchy.
- With a command line and the web interface extension:
  1. On the child server, open a command line shell.
  2. Go to the directory where **`rbagent`** is located.
  3. Run `rbagent rad-srvsync`.

  Note: The `rbagent rad-srvsync` command has the following usage:

  `rad-srvsync` `[cascade] [children]`

  The command used without options, triggers a file replication on the server it connects to (local server). `cascade` is an optional keyword that triggers a file replication on the local server and recursively on all the servers below the local server. `children` is an optional keyword that triggers a file replication on all the children of the local server (just one level).

Now, you can replicate the children of this OS deployment server. You can also setup a replication schedule.

Replicating offline with the web interface extension

Replication with the web interface extension (RbAgent) relies on the `sync.pak` package. This package must be located with the other `.pak` files, in *`INSTALLDIR`*`\packages`, on both parent and child servers. The compiling process generates and stores the `sync.pak` package under the *`INSTALLDIR`*`\inactive_packages` directory, where:

INSTALLDIR
Is, for example, `C:\Program Files\Common Files\IBM Tivoli` on Windows platforms.

The main concept behind this replication process is to keep a list of important parent server states and to create differentials between states. These differentials can then be transferred from parent to child to update the child server.

Attention: Between the creation of a checkpoint and the end of the creation process of the corresponding differential, steps 1 and 2 of the procedure, the objects on the parent server must not be modified. It is prohibited to create or modify any deployment object including, but not limited to, system profiles, software modules, deployment schemes, and hardware configurations.

Note: When you perform a replication using `sync.pak`, for example creating a differential `.rad` file, the `.rad` file gets renamed to `rad.ok`. But the server status under **Server parameter** > **Server replication** is still yellow because the value used to display the yellow triangle is not updated by `sync.pak`.

To perform replication with RbAgent:

1. Create a new checkpoint on the parent server when it is in a stable state. A new checkpoint must be created after major changes on the parent server. Checkpoint `0` (zero) refers to the initial state of the server and is always present. For more details about how to create a check point, see "Specific RbAgent commands" on page 337
2. Create a differential between a chosen checkpoint state and the latest checkpoint state of the parent server. This builds a `.rad` file (or several `.dat` files if you have indicated a file size limit) in the `TPMfOS Files\import` directory.

   You can perform this step synchronously (RbAgent waits until the task is complete before returning control) or asynchronously (RbAgent returns control immediately). In the asynchronous mode Tivoli Provisioning Manager for Images prevents you from launching two `.rad` file creation processes concurrently.

   Note: If changes have been made on the parent server since the last checkpoint, you cannot create a differential with the last checkpoint as endpoint. You must first create a new checkpoint reflecting the current state of the parent server.
3. Transfer the `.rad` file from the parent server to the child server. Tivoli Provisioning Manager for Images does not interfere in this transfer process.
4. To replicate your child server, copy your differential file from its current location (either the parent server or a local directory) to the specific `TPMfOS Files\import\auto` directory. This directory is automatically created when the `sync.pak` package is present. Tivoli Provisioning Manager for Images checks for changes in the `TPMfOS Files\import\auto` directory automatically. Whenever a new file is found, it is checked for coherence (if it is a `.rad` file), or recomposed as a `.rad` file (if it is a series of `.dat` files). The file is renamed with a `.ok` extension if the process succeeded, or with a `.err` extension in case of error.
5. The contents of the `.rad.ok` file are automatically replicated with the shared repository if the checking process is successful.

The checkpoint-based replication ensures that server files are up to date. This is enough if both OS deployment servers share the same database. If the OS deployment servers are using separate databases, it is necessary to replicate the database records as well. This can be achieved through export files as well if needed, using the web interface extension command-lines **`rad-exportdb`** and **`rad-importdb`**.

You can customize the files that are replicated by indicating which folders are concerned. To do so, edit the `[RSyncConf]` section of the `TPMfOS Files\global\serverstate\sequence.ini` file where the list of folders has been initially populated. Subfolders are recursively and automatically included.

Replication with the web interface extension

The replication process has been redesigned for improved performances in branch office scenarios. Instead of file copies, replication of shared repository files is possible with the web interface extension (RbAgent) and a specific package implementing specialized command-lines for RbAgent (`sync.pak`).

The load on the parent server is reduced. Control over the performance of the replication process is split into operations on the parent and those on the subordinate server. The parent server does not must be running when a subordinate server is replicating itself.

Specific RbAgent commands

The `sync.pak` package implements several RbAgent commands that you must use for this specific replication process. With these commands, you can export and import a database content, create new checkpoints, list existing ones, and create `.rad` files.

Database replication commands

`rad-exportdb filename`
: Exports a RAD file named `filename`. This command dumps a single file of all database records that describe the deployment objects in a server at a given time. The file name is the only argument that is required to export the database. The database is exported to `global/rad/`.

`rad-importdb filename`
: Imports a database dump file. This command imports the database dump file generated by **`rad-exportdb`** on a remote server so that the local server can be upgraded to the exact same content. The file name is the only argument that is required to import the database. The file must be located in `TPMfOS Files/global/rad/`.

File replication commands

`sync-seqidlist`
: Returns the list of all valid checkpoints. These checkpoints are extracted from the server file system. The command typically exits with the status `0`. If the command exits with status `1`, an error has occurred and is described in the standard output.

`sync-newseqid new-sequence-id | auto [force ] [TaskID=n Description=d]`
: Creates a new checkpoint. `new-sequence-id` is a string identifying the new checkpoint; `auto` is the keyword that generates a new sequence ID automatically; `force` is an optional keyword that overrides an existing checkpoint; `n` is an unsigned 64-bit integer in decimal form used for status reports; `d` is a freely usable string, used for status reports. The command typically exits with the status `0`. If the command exits with the status `1`, an error has occurred and is described in the standard output. Checkpoint information is stored in `TPMfOS Files/global/serverstate`.

`sync-radget newdiff.rad from-seqid | 0 [-split n ] [TaskID=m Description=d ]`
: Synchronously creates a differential RAD file. `newdiff.rad` is a RbAgent URL. For example, `local://root/c$/temp/diff-0-1.rad`; `from-seqid` is the reference checkpoint from where files can be omitted; 0 is the initial checkpoint; `-split n` optionally forces splitting the file into fragments of `n` MB. `m` is an unsigned 64-bit integer in decimal form used for status reports; `d` is a freely usable string, used for status reports. The command typically exits with the status `0`. If the command exits with the status `1`, an error has occurred and is described in the standard output. The command creates a `newdiff.rad` file. With option `Split`, several files can be created. They are automatically renamed. For example, `newdiff.rad` becomes `newdiff-rad-x-of-y.dat`. Each fragment finishes with an MD5 and a signature (20 bytes). With the option `Split`, `newdiff-rad.dsc` is a description of the fragments. The command cannot start if the server files do not match the last checkpoint. Running `sync-newseqid` before `sync-radget` is a prerequisite.

`sync-srvradget newdiff.rad from-seqid | 0 [Split=n ] [TaskID=m Description=d ]`
: Asynchronously creates a differential RAD file. This is the asynchronous version of the `sync-radget` command. Another important difference is in

the definition of the parameter `newdiff.rad` which is here a path relative to `c:\TPMfOS Files\import`. If the command returns after several minutes, the OS deployment server is not responding. Although asynchronous, two or more `sync-srvradget` commands cannot run concurrently.

Replicating one time in command line

In some cases, you must replicate your server once, for example if you want to change your server hardware or your database, or if you want to make a copy of a production server to run tests on it.

One time replication is available from a parent version equal or lower than the child version, namely

- from version 5.1.1 interim fix to version 5.1.1 interim fix,
- from version 5.1.1 interim fix to version 7.1
- from version 7.1 to version 7.1.
- from version 7.1 to version 7.1.1
- from version 7.1.1 to version 7.1.1
- from version 7.1.1 to version 7.1.1.1

You must be aware that one time replication deletes the content of the server on which the operation is performed. Take all precautions to ensure that you have nothing valuable on the OS deployment server from which the command is run or on its database.

The command line must be run on the receiving (child) server.

1. Open a command line shell.
2. Go to the directory where **`rbagent`** is located.
3. Run `rbagent rad-replicate <parent-ip>` where `<parent-ip>` is the IP address of the parent server you want to replicate.

Now, you can use your newly replicated server to run tests without impacting your production server, or to replace obsolete hardware.

Server replication status and logs

You can see the server replication status from the Web interface. Go to **Server** > **Server parameters** > **Server replication**. The icons on this page inform you visually of the replication status of your servers. Logs also contain information about the replication process.

Status

- On the lower left hand-side of a server icon, the **up/down indicator** is displayed. The status indicator can take two different values:
  - A blue circle with a light center, indicating that the OS deployment server is up and running.
  - A black dot, indicating that the OS deployment server is down.

- On the lower right side of a child server icon, the **replication status indicator** is displayed. The status indicator can take three different values:
  - ✗

    A cross in a red dot indicates that the selected child server is not up-to-date with its parent. Files are missing on the child server; the child server must be replicated with its parent before any action is performed.
  - ⚠

    A yellow triangle with an exclamation point indicates a warning. A discrepancy was discovered between the child and the server files. Some files can have been updated or added on the parent server.

    Click **Object version** to view which deployment objects are not up-to-date. If you plan to use any of these objects, you must replicate your server first. This page contains yellow triangles if SSL is not disabled or if the servers are temporarily unresponsive.
  - ✓

    A green dot with a white check mark indicates that the child server is up-to-date with its parent.

Note: When a server is down, it keeps the replication status indicator it had when it was last running. Replicating while a server is down is not possible.

Logs

Whether server replication is activated manually (using the replication link in the web interface or using the web interface extension), through scheduling in the web interface, or with the `config.csv` file, server replication corresponds to a set of tasks. Several logs are available to monitor the replication process and these tasks.

Sync log file or `files/logs/sync.log`

This log file contains information specific on replication: checking files, finding them or not, copying them, and so on. Its content can be viewed in the **Server log files** page of the web interface.

`files/global/hostactivities`

This directory contains the list of all target tasks. The content is merged with the information provided by `activities.log` and can be viewed from the **Tasks** page of the **Server history**: select the task and choose **Show log file**.

Switching from an ODBC to a JDBC gateway

In a server hierarchy with heterogeneous databases, the replication process works only if you use the JDBC gateway instead of the ODBC gateway.

To switch from an ODBC to a JDBC gateway, you must perform the following steps:

1. Create a database and ODBC source.
2. Install Tivoli Provisioning Manager for Images.
3. Run `net stop remboserver` to stop both server and database gateway.

4. Run regedit to modify the registries:
   a. Delete the **SubService** entry that starts the ODBC gateway. An example is:
      ```
      [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
      RemboServer\Parameters\_SubServices]"RemboODBC"=dword:00000000
      ```
   b. Add the **SubRun** entry that automatically starts the JDBC gateway. An example is:
      ```
      [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
      RemboServer\Parameters\_SubRun]
      "dbgw"="\"C:\\Program Files\\Java\\jre1.5.0_11\\bin\\java.exe\" -Xrs
        -cp \"C:\\Program Files\\Common Files\\IBM
        Tivoli\\dbgw.jar;C:\\Program
        Files\\derby\\db-derby-10.2.2.0-bin\\lib\\derbyclient.jar\"
        -Djdbc.drivers=org.apache.derby.jdbc.ClientDriver
        com.rembo.dbgw.Dbgw"
      ```
5. Create or update the `config.csv` file to use the JDBC syntax.
6. Copy all jars needed for accessing the remote databases into the `..\\Common Files\IBM Tivoli` directory.
7. Run `net start remboserver` to start both server and database gateway.
8. Check that the JDBC gateway is started automatically according to the **SubRun** registry entry.

Removing an OS deployment server from the hierarchy

Removing an OS deployment server from a multiserver hierarchy depends on whether you are using a single database or a multiple databases infrastructure.

- In a single database infrastructure

  Note: Removing a running OS deployment server from the database in a single database infrastructure can lead to unknown side-effects.
  1. Make sure the OS deployment server you want to remove does not have any child. If it has any, assign it a new parent.
  2. Go to **Server** > **Server parameters** > **Server replication**.
  3. Select the child to be removed.
  4. Click **Make this OS deployment server a standalone OS deployment server**.
  5. On the former child, run the `rbagent rad-resetscope` command to set the scope of all the objects on the server to local. Otherwise all replicated objects remain read-only.
- In a multiple database architecture, with a running OS deployment server to be removed
  1. Edit the `config.csv` file of the OS deployment server you want to remove. Change the *MasterIP* value to `SELF` and *AutoSync* to an empty string. Do not change the value of *MasterDbName*.
  2. Restart the OS deployment server. The OS deployment server sees the changes in its `config.csv` files and updates the database of its former parent to indicate that it is not its child anymore.
  3. On the parent server, go to **Server** > **Server parameters** > **Server replication**.
  4. Select the child being removed and click **Make this OS deployment server a stand-alone OS deployment server**. The child moves to the **Standalone OS deployment servers** section.
  5. Select the child again and click **Remove this OS deployment server from database**.

6. On the former child, run the `rbagent rad-resetscope` command to set the scope of all the objects on the server to local. Otherwise all replicated objects remain read-only.

   The OS deployment server is now detached from its parent but it keeps any child it might have had.

- In a multiple database architecture, when the OS deployment server to be removed is not working (if it crashed, for example)
  1. Make sure the OS deployment server you want to remove does not have any child. If it has any, assign it a new parent.
  2. Go to **Server** > **Server parameters** > **Server replication** on an OS deployment server parent to the one that you want to remove.
  3. Select the OS deployment server to be removed and click **Make this OS deployment server a stand-alone OS deployment server**. The child moves to the **Standalone OS deployment servers** section.
  4. Select the OS deployment server again and click **Remove this OS deployment server from database**.

  The information about the OS deployment server is removed from the database of its parent.

If the OS deployment server you have removed does not have any child, you can safely shut it down or uninstall it.

Chapter 9. Security

This section provides the user with information regarding security issues.

Security roles and access to the Web interface

Security roles allows you to create groups of users with restricted privileges to access the web interface, thus enhancing the overall security of your OS deployment server.

Administrator name and password

There is a unique administrator name and password for each OS deployment server. These name and password must be used first to create an authentication domain and create security roles. Afterwards, they must be stored in a safe place for reference. Instead, users must type their own user names and passwords to connect themselves to the web interface, as defined in the *HTTP* authentication domain.

Security roles

Security roles allows you to define groups of users with specific privileges on the OS deployment server. For each role, you can define which pages of the web interface they can view, which administrative groups of targets they can act upon, and which tasks they are allowed to perform. Preexisting roles are *Administrators* and *Operators*. However, any role can be created. Each user must be assigned to one or several security roles.

Note: Users belonging to several roles cumulate their privileges. You must therefore edit the two predefined roles which give overall control to all users.

HTTP authentication domain

Authentication domains determine how user and password information are verified, either locally or remotely. Users authenticated in the *HTTP* authentication domain and which belong to a security role gain access to the web interface, according to the privileges of the role. The specific *HTTP* authentication domain is a prerequisite to create security roles.

Connections to the web interface

You can monitor who is connected to the web interface on the page **Server** > **Server status** > **Network connections** page, under **Web interface sessions**.

Example

- *John* is a user of the computer on which the OS deployment server is installed.
- The *HTTP* authentication domain is set to local. Therefore, users trying to connect to the web interface must be users of the computer on which the OS deployment server is installed.
- A security role called *Rome Operators* with restricted privileges has been created. Members of this role can view all the pages of the web interface, but only the

targets belonging to the *Rome* administrative group. Moreover, they are denied any action which would change deployment objects or server parameters.

- *John* is assigned as a member of the security role *Rome Operators*.

In this configuration, user John can log into the web interface using his local password. Once logged in, he can deploy targets from the *Rome* administrative group. But he needs to ask an administrator if he wants to create a new software module and bind it to a profile, as he does not have the necessary privileges.

Creating an HTTP authentication domain

The *HTTP* authentication domain is a prerequisite for using security roles to control access to the OS deployment server through the web interface.

1. Go to **Server** > **Server parameters** > **Predefined channels**.
2. Click **New auth. domain**.
3. Type `HTTP` as domain name. Case matters.
4. Select the type of domain you want. There are three possibilities: local, remote NT and RADIUS.
5. Optionally, enter a user group to restrict access only to the users of this group.
6. Click **Save**.

You can now create security roles.

Creating security roles

Creating security roles allows you to provide access to the web interface for users besides the administrator, to restrict access to some pages and some features, and monitor who is currently logged in.

Before you can create valid security roles, you must have created an *HTTP* authentication domain.

1. Go to **Server** > **Server parameters** > **HTTP console security**.
2. Click on **New security role**.
3. Provide a name for the new role.
4. Edit the role parameters.
   a. Deselect web interface pages to which role members must not have access.
   b. Deselect administrative groups to which role members must not have access.
   c. Select features that you want to deny access to.
5. Click the available links to remove from and add members to the role. When entering a user or group name, it must correspond to a name which can be identified in the HTTP authentication domain.

   Note: Users who are members of several roles cumulate the privileges of all their roles.
6. Click **Save**.

You can now log into the web interface using the username and password of a role member.

To edit the role at a later time, to change privileges, or to add or remove members, go to **Server** > **Server parameters** > **HTTP console security** and click on the role name.

Changing the administrator password

To change the server administrator password you have two different options.

Modifying a server administrator password impacts on communication with all components. For example, if you have a synchronized server or rbagent components running, you must update their passwords as well.

To change the server administrator password, perform one of the following actions:

- On the server console, go to **Server** > **Server Parameters** > **HTTP Console security** and click **Edit** in the yellow header to modify the administrator password.
- Modify the `rembo.conf` file as follows:
  1. Export your current server parameters by clicking **Export Configuration** on the server console. An updated `rembo.conf` file is provided.
  2. Modify your `rembo.conf` file with any text editor.
  3. Stop the Rembo service or deamon.
  4. Reload the configuration with the command:
     ```
     rembo -d -v 3 -c rembo.conf -exit
     ```
  5. Restart the service or deamon.

You can use plain text or MD5 encrypted passwords. Ensure that you enter only the MD5 string, without any line feed or carriage return.You must change the `rbagent.conf` files located on all computers running the Rembo agent so that the Rembo agents can contact the server.

Backups of server files

Tivoli Provisioning Manager for Images operating system images and other files are stored in a folder, and are accessible using file browsing tools. All of these files are stored in the data directory (typically `C:\TPMfOS Files` for Windows).

This directory contains regular files and control files, with `.md5`, `.dir` and `.inodes` extensions. These special files contain information about the file system structure, including the internal file number used by the provisioning server to identify a file.

When backing up files, it is important that you include both regular files and the provisioning server special files. When restoring files (or adding individual files), the provisioning server automatically detects new files and creates associated control files. Adding a file does not necessarily mean that the file will be usable by the provisioning server. A database entry is typically needed to describe what the file is used for. Therefore, it is also crucial to backup the database at the same time you backup the server files.

If you only want to back up a specific deployment scheme, system profile, or software module, it is easier to use RAD files.

Importing and exporting RAD files

Tivoli Provisioning Manager for Images allows you to export and import different types of objects if has created.

Some objects are exported and imported imbedded in a *RAD file*, others in a *target list*.

RAD files (with a `.rad` extension) can contain a single object or multiples objects. A RAD file can contain:

- WinPE deployment engines
- deployment schemes
- hardware configurations
- software modules
- system profiles

With RAD files, you can:

- move objects between OS deployment servers that do not have a good network connection between them
- archive objects without a running OS deployment server

Note: A RAD file can only be imported on a computer with the same byte order (little endian or big endian) as the computer on which the OS deployment server or web interface extension used for the export was installed.

Importing RAD files

To import RAD files in an OS deployment server:

1. Click **RAD Import** on the **Task templates**, **Profiles**, or **Software modules** page of the web interface.
2. Follow the instructions of the RAD Import Wizard.

   Note:

   a. When selecting the objects you want to import from the RAD file, you have the option to import the software application order (**Software stages**). Use this option carefully as the imported software application order overwrites the order present on your OS deployment server.
   b. When importing deployment objects from a RAD archive, the byte ordering of the importing server must be the same as the one used by the exporting server. To be able to import a RAD archive created on a server using a byte ordering different than the importing server, you must perform the import using the web interface extension, running on a platform with the original byte ordering.
   c. When importing a Windows 2008 or Windows Vista system profile in a RAD file created with version 7.1.1.2 of the product, you also need to import the corresponding WinPE 2 ramdisk. If you do not, your system profiles cannot be upgraded and your system profiles are tagged as *too old*. If the necessary WinPE 2 ramdisk is not present in the RAD file, you need to export it again from your 7.1.1.2 OS deployment server.
   d. When importing a WinPE deployment engine or a system profile, you also import the associated driver bindings. A check is then performed to find the associated driver software modules.

      The driver software module is present in the RAD file
      The binding is fully restored.

      The driver is found on the OS deployment server
      If the driver software module is not in the RAD file, a search is performed on the OS deployment server to look for the software module.

      The search is performed on the software module ID. Therefore, a match occurs only if the software module was exported from the same server. In some rare cases, the driver software module cannot be found although it is present because its ID has changed.

If the driver software module is located at the time the WinPE deployment engine or the system profile is imported, the binding is fully restored.

This implies that driver software modules must be imported before, or at the same time as, WinPE deployment engines and system profiles for the bindings to be restored.

The driver is not found

If the driver software module is found neither on the RAD file, nor on the OS deployment server, the driver binding cannot be restored.

Exporting RAD files

To export a RAD file:

1. Click **RAD Export** on the **Task templates**, **Profiles**, or **Software modules** page of the web interface.
2. Follow the instructions of the RAD Export Wizard.

- When exporting a RAD file, the software application order is automatically included.
- When exporting a RAD file, driver bindings associated with exported WinPE deployment engines and system profiles are automatically exported. However, the software modules associated with these bindings are not exported, unless they have been individually selected.
- When deciding where to generate the RAD file, be aware that the option to **download it directly from the server** is not available if the estimated size of the `.rad` file is bigger than 2GB, because of web browsers limitations.

  Note: If you export a RAD file by running the rbagent **`rad-radget`** command from a remote machine different from the OS deployment server and with the web interface extension installed, ensure that the local temporary directory, where the RAD file is temporarily copied, has enough space. For example, if the rbagent runs on UNIX, and the `/tmp` directory does not have enough space to contain the RAD file, then the **`rad-radget`** command fails. To avoid this problem you can either add more space to your temporary directory or change the temporary directory, as follows:

  1. Stop the OS deployment server.
  2. Define a new temporary directory. For example on UNIX, enter: **`export TEMP=/root/temp`**.
  3. Start the OS deployment server and enter the rbagent **`rad-radget`**command again.

Importing and exporting targets lists

A target list file is a `.csv` text file with comma-separated values. Importing a target list is useful for adding large numbers of targets to the OS deployment server without having to start them individually on the network. You can also import a PCI inventory for a single target in an `.ini` file.

Familiarize yourself with target lists and PCI inventory.

Target list

Before you can import a target list, you must either export one or create a new one.

Information about each target in a target list is a collection of more than seventy items, including:

- MAC address
- IP address
- User parameters
- Motherboard information
- Processor information

To view the complete list of items, export a target list and read the beginning of the `.csv` file.

For the OS deployment server to successfully import targets in a list, you must fill in at least one of the following items:

- Serial number
- MAC address
- UUID
- IP address

The item you fill in can vary from target to target. Other items can remain empty.

Target lists above 1 GB in size cannot be imported into an OS deployment server because of browser limitations. Therefore, you cannot use target lists for more than about 1000 targets.

Note: Do not use target lists to back up target information. To back up target information, you must back up the database with an appropriate tool. Lists of targets are not as complete as the database. In particular, target lists do not include some crucial target information found in the database, for example:

- Bindings
- Disk inventory
- PCI inventory
- Deployment history

PCI inventory

You can export a PCI inventory to a USB key or disk. They can be booted through a network boot media, without having network drivers.

- **Importing a target list**
  1. Go to the**Target Monitor page** in the web interface.
  2. Click **Import targets**.
  3. Indicate the location of the `.csv` file.
  4. Click **Ok**.
- **Exporting a target list**
  1. Go to the**Target Monitor** page in the web interface.
  2. Click **Export targets**.
  3. Click **Save**. You can change the default file name, `hostexport.csv`, and location.
- **Importing a PCI inventory**
  1. Go to the**Target Monitor page** in the web interface.
  2. Click **Import targets**.
  3. Indicate the location of the `newhost.ini` file.

4. Click **Ok**.

Exporting and loading configurations

You can export the configuration of the OS deployment server or load configuration settings that you have previously exported.

Exporting a configuration

To export the current configuration of your OS deployment server, click **Export configuration**. The button and the contextual menu item are present on the following pages of the web interface:

- **Server > Server parameters > Configuration**
- **Server > Server parameters > HTTP Console Security**
- **Server > Server parameters > Predefined channels**

Loading a configuration

To load a server configuration stored in the `rembo.conf` file, follow these steps:

1. Stop the OS deployment server.
   - On Windows operating systems, type `net stop remboserver` in a DOS window.
   - On UNIX operating systems, see "Startup scripts" in the Installation Guide.
2. Load the new configuration by typing
   ```
   rembo -d -c rembo.conf -exit
   ```
3. Restart the database gateway and OS deployment server.
   - On Windows operating systems, type `net start remboserver` in a DOS window to start both services.
   - On UNIX operating systems, see the Installation Guide.

Fault tolerance

A system is fault-tolerant if it can continue to perform despite parts failing. Fault tolerance helps to make your remote-boot infrastructure more robust.

In the case of OS deployment servers, the whole system is fault-tolerant if the OS deployment servers back up each other. When a server fails, other servers handle the requests from the down server.

Implementing fault tolerance at the Tivoli Provisioning Manager for Images level does not mean that your whole network infrastructure is fault-tolerant. You can implement fault-tolerances at all levels:

- At the physical level, by having redundant power sources (if all OS deployment servers are out of power at the same time, fault-tolerance at the product level is useless)
- At the network level, by having backup network links, and backup active elements (the backup server must be able to reach remote-boot targets)
- At the network operating system level, by having multiple network domains, or by running OS deployment servers outside of your domain architecture (OS deployment servers should not be all linked to the same NT PDC, or the same NFS server)
- At the DHCP level, by having multiple DHCP servers on the same subnet
- At the Tivoli Provisioning Manager for Images level, by implementing the fault-tolerance instructions.

- At the operating system level. If Tivoli Provisioning Manager for Images is able to survive to a severe problem, but then the operating system cannot find its network server, fault tolerance is useless

The following sections present information about how to implement fault tolerance at the DHCP and Tivoli Provisioning Manager for Images levels. Other levels are beyond the scope of this document.

Fault tolerance at the DHCP level

The DHCP protocol allows the implementation of fault tolerance and load-balancing very easily. If you connect two DHCP servers to the same IP subnet, and both servers are configured to serve IP addresses on this subnet, the protocol handles all conflicts between the two servers. A system is fault-tolerant if it can continue to perform despite parts failing. Load balancing specifies the maximum number of DHCP/BINL requests to a OS deployment server in one minute.

When a remote-boot target requests an IP address, the request packet is sent to the local broadcast address, that is, to all targets connected to the same IP subnet as the remote-boot target. If one or more DHCP servers are connected to the subnet, they send a DHCP offer packet to the remote-boot target, containing an IP address that has either been allocated in the server pool or administratively assigned to the remote-boot target (in case of statical binding between the hardware address and an IP address in the DHCP configuration, also called *a reservation*). If more than one DHCP offer packet is received by the remote-boot target, only the *most informative* offer is kept by the target.

When the remote-boot target has selected a valid offer, it replies to the server from where the offer originated with a broadcast packet. This packet is received by all the targets connected to the local subnet, including the DHCP servers. This packet is used by DHCP servers to know if their offer was accepted or refused by the remote-boot target. If the target accepts, the IP address is locked in the DHCP server database, and the DHCP process can continue in unicast mode between the remote-boot target and the DHCP server. If the reply is for another offer, the server releases the IP address for its offer (which has been ignored by the target), and locks the IP address seen in the offer reply, to mark the IP address as used on this local subnet (even if the IP address has been allocated by another DHCP server).

Because of this, you can implement fault-tolerance by configuring multiple DHCP server for the same subnet. If the DHCP servers are identically configured, then the remote-boot targets always select the offer coming from the fastest server. Use this offer to implement load-balancing at the same time as fault-tolerance: the fastest server is always selected, and if the fastest server becomes overloaded, another server can send its offer first, and then it becomes the fastest server.

Fault tolerance at the Tivoli Provisioning Manager for Images level

Fault tolerance helps to make your remote-boot infrastructure more robust. A system is fault-tolerant if it can continue to perform despite parts failing. Fault tolerance at the product level is implemented with two configuration parameters: `Backup` and `BootReplyDelay`.

The boot process is made of several phases:

- DHCP discovery

- PXE discovery
- MTFTP download
- The product

Understanding these phases is key to understanding fault tolerance. Fault tolerance at the DHCP level is described in the previous section. You can implement fault tolerance at the PXE discovery level by using multipleOS deployment servers in Proxy DHCP mode (there is no OS deployment server on the DHCP server target, but all OS deployment servers are connected to the subnet). In Proxy DHCP mode, OS deployment servers send PXE reply packets to DHCP discovery packet initially sent by the remote-boot target. Because DHCP discovery packets are sent to the broadcast address, all OS deployment servers receive the discovery, and all send a reply packet, with the following considerations:

- The remote-boot target must be known by the OS deployment server (either by being a member of a target group in the server OS configuration, or if the OS configuration allows unknown targets to connect);
- The server does not answer immediately if the parameter `BootReplyDelay` is set.

You can use `BootReplyDelay` to introduce a preference order between the OS deployment servers on a same subnet. The server with the lowest `BootReplyDelay` is the first to answer DHCP discovery packets. All remote-boot targets are redirected to this server. If this server fails, the server with the second lowest value for `BootReplyDelay` then answers, and so on. Fault tolerance at the PXE discovery level is in place.

If several OS deployment servers have the same value for the `BootReplyDelay` parameter, they all send the PXE reply at the same time, and the remote-boot target selects the fastest server. This specific environment implements load-balancing at the product level.

When the remote-boot target has selected its DHCP and PXE servers, the product bootstrap downloads from the PXE server (OS deployment server), and the target side is started. You can implement fault tolerance inside the product by using the `Backup` parameter for specifying a backup server. This value is sent to the remote-boot target during the initial startup of the target computer, and is used as a backup server if the primary server fails. The internal network protocols used in the product have been designed to enable the target to switch from the primary to the backup server in the middle of a file transfer. This only works under the following considerations:

- Files opened in write mode (upload to the product) cannot switch to a backup server. This could corrupt data on the OS deployment server, because one part of the file could be written on the primary server, and the other part on the backup server.
- The file system structure on the primary and backup servers must be strictly identical (that is, the same content under the `files` directory of the server).

Use backup servers at the product level (with the `Backup` parameter) when you stabilize your system (hard disk images are built, scripts are ready). After you stabilize the primary server, copy the `files` directory from primary to backup server, and set the `Backup` parameter on primary server.

Network security constraints

In many enterprise environments, an administrator must consider network security constraints.

For example, some ports can be unavailable to secure network traffic in and out of the enterprise.

By default, Tivoli Provisioning Manager for Images uses the following ports on the OS deployment server for communication:

- *DHCP* : port 67 *UDP*
- *PXE BINL* : port 4011 *UDP*
- *TFTP* : port 69 *UDP*
- *MTFTP* : port 4015 *UDP*
- *NBP* : port 4012 *UDP*
- *FILE* : port 4013 *UDP & TCP*
- *MCAST* : port 10000-10500 *UDP* Address: 239.2.0.1-239.2.255.255
- *HTTP (web interface)* : port 8080 TCP
- *HTTPS* : 443 TCP
- *Database gateway* : port 2020 TCP

On *targets*, the default ports are:

- *DHCP* : port 68 *UDP*
- *MTFTP* : port 8500-8510 Address: 232.1.0.1 *UDP*
- *MCAST* : port 9999 *UDP*
- *Remote control (web interface extension)* : port 4014 *UDP*

All of these ports can be modified, with the exception of port 69 for TFTP. Port 69 is part of the PXE specification, independent from Tivoli Provisioning Manager for Images, and cannot be modified. Any product using PXE boot needs to have this port open to permit PXE boot. This port needs to be open only on the OS deployment server, not on the target computers.

If you must modify ports (server or target) to conform to your network security constraints, you can use the web interface or edit the `rembo.conf` configuration file (and stop and restart your OS deployment server with option `-c rembo.conf`).

Make sure the necessary ports are open in both directions on both the OS deployment server and the targets to use all the features of the product. For example, to use multicast, *MCAST* and *MTFTP* ports, among others, must be open in both directions on the OS deployment server and on the targets.

Note: If you do not want to use PXE to remote boot targets, you can create a network boot media.

Avoiding new security breaches

After you have installed the OS deployment server on your network while taking into account network security constraints, you still must ensure that using Tivoli Provisioning Manager for Images does not create new security breaches.

1. Protect your network against rogue PXE servers that can have access to your network. Otherwise, target computers can boot on the rogue server instead of the legitimate PXE server.
2. Prevent unwanted target computers from booting on your PXE server, unless you want to risk transferring sensitive information to unsecure computers.

Rogue PXE servers

A rogue PXE server is a server on a network which is not under the administrative control of the network staff.

By default, the PXE protocol is not protected against rogue PXE servers when it is working in boot discovery mode. There are ways to prevent this type of breach.

The target sends broadcast packets to the network requesting a PXE answer. The first PXE server to respond to the request takes control of the target computer. A rogue PXE server answering the request faster than the legitimate Tivoli Provisioning Manager for Images PXE server can take control of computers booting onto the network.

Using PXE in boot discovery mode is a well known security breach, independent from Tivoli Provisioning Manager for Images. While DHCP discovery must broadcast requests (the target does not yet possess any network information), there are ways to prevent the PXE security breach and permit only authorized PXE servers to answer requests from targets.

Using DHCP options to close the breach

Deactivate boot discovery mode for PXE targets. After this is done, computers trying to contact a PXE server must know the specific address and can no longer send broadcast packets. Information is transferred at the DHCP stage, by using options 60 and 43. Using these options, the DHCP server returns the target its IP address and the IP address of the authorized PXE server. If necessary, option 43 can contain several IP addresses for backup servers.

information about how to configure the DHCP server is located in the Tivoli Provisioning Manager for Images7.1.1.14 Installation Guide, Chapter 4.

Unwanted target computers

You must ensure that target computers are legitimate in order not to distribute sensitive information outside of appropriate computers.

To achieve this with Tivoli Provisioning Manager for Images, set the OS deployment server to closed mode. In closed mode, the OS deployment server does not accept new targets and sends boot information aboutly to the targets listed in its database. To activate this parameter,

1. Go to **Server** > **OS deployment** > **Task templates**.
2. Select **Idle Layout** and double click **Idle state**.
3. Click **Edit** for the section **Handling of unknown targets**.
4. Select **Completely ignore unknown targets (closed OS deployment server)**

   Note: When you perform step 4 you also close the server which does no longer allow the registration of new rbagents.

In closed mode, the OS deployment server (PXE server) checks the MAC address and the IP address of potential targets and sends the Tivoli Provisioning Manager for Images bootstrap only if these addresses belong to a known target.

The bootstrap is sent by TFTP, which is a non-secured protocol. However, this bootstrap is very small (around 300 KB) and does not contain any critical information. After the bootstrap runs on the Tivoli Provisioning Manager for

Images target, all other transfers are performed using secure protocols. When the bootstrap is in place, the OS deployment server checks the UUID and the serial number of the Tivoli Provisioning Manager for Images target for before transferring any other data. This ensures that the MAC and IP addresses to which the bootstrap was sent were not faked.

If no rogue server can interfere between your targets and the OS deployment server and no unknown target can boot from your OS deployment server without authorization, you have ensured that Tivoli Provisioning Manager for Images does not add security breaches to your network environment.

Security issues and the web interface

Sessions on web interface have been made as secure as possible. However, security relies also on users and the way they use the product.

To ensure the highest possible level of security using the web interface, the following features have been implemented:

- Connections are made using the encrypted HTTPS protocol.
- Sessions on the web interface are identified by a unique session identification number. If you need a second web interface, you must log on to a new one to ensure having distinct session identifiers. If you open a new window or cut-and-paste the URL without logging in again, it can result strange web interface behavior.
- Sessions expire after a given delay, automatically logging users out if they forget to click **Logout** at the end of their session. To modify the length of this delay, see web interface parameters .

Chapter 10. Booting targets without using PXE

For BIOS targets, if you do not want to use PXE on your network, you can use Tivoli Provisioning Manager for Images to create a network boot CD, DVD, or USB drive. For UEFI targets, only PXE booting is supported.

With network boot media, your target can boot and connect to the Tivoli Provisioning Manager for Images server in a PXE-less environment. Use this kind of deployment when it is not possible to use PXE to boot the target.

Some typical situations are network card without PXE support, firewalls preventing PXE traffic, non-allowed PXE boot, or an unavailable DHCP server. In particular use the original Windows PE-based network boot CD to minimize target hardware compatibility issues.

To create the network boot media, you can either use the wizard or run command lines from a computer with the web interface extension installed. You can create network boot media in both kernel mode or in kernel-free mode.

Note:

- Network boot media must be updated every time the OS deployment server is updated or upgraded to ensure compatibility with the OS deployment server.
- If your network boot media is optimized for Windows operating systems, you must create the media from an OS deployment server or a web interface extension installed on a computer with the same byte order (little endian or big endian) as the one on which you want to use the network boot media.
- You cannot capture hardware information or deploy a hardware configuration from a target started with a network boot media.
- Before you create your network boot media, make sure that the **Disable DHCP/BINL module** parameter is set to **no**.

Creating a kernel-based network boot USB drive with the wizard

Tivoli Provisioning Manager for Images can automatically generate bootable USB drives that connect the target to an OS deployment server, without using DHCP or PXE, to perform deployments.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must be formatted as FAT32 or NTFS. USB keys already filled with a bootable operating system might not work.

Note: SuSE Linux Enterprise Desktop cloning is not supported on USB drive deployments

These bootable USB drives can also be used to deploy computers without a PXE compliant network adapter.

To create OS bootable USB drives:

1. Perform one of the following steps:
   - Go to **Server** > **OS deployment** > **Task templates**.
   - Go to **Server** > **OS deployment** > **System Profiles**.

- Go to **Server** > **OS deployment** > **Software modules**.

2. Click **Generate media**.
3. Select **Create a network boot USB key** to start the USB key wizard. Click **Next**.
4. Specify the operating system on which to boot the target. Select **Linux** to load a Linux deployment engine environment, **Windows** to load a WinPE deployment engine, or **Both** to have the two.

   Note:

   - If you use a network boot media to deploy Linux profiles, you cannot use the HTTP protocol. For this reason when you create a deployment scheme ensure you do not select the **Download files with a network share when applicable** option in the **Network settings** section.
   - If you use a network boot media and want to erase hard disk content, your media must contain a WinPE deployment engine. Therefore, you must select either **Windows** or **Both**.
5. If you have selected **Windows** or **Both**, and if you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. If you want to obtain the target IP address through DHCP, select **Dynamic IP address with DHCP**, and click **Next**.

   If you want to use a fixed IP address for your target instead of having it go to the DHCP server, select **Static IP address**, and click **Next**.

   a. Enter the target IP address, gateway, and network mask.
   b. (*Optional*) Select **Allow IP address override at runtime** to be able to modify the target IP address when starting up the target.
   c. Click **Next**.
7. Enter the IP address of the OS deployment server.
8. (*Optional*) Select **Allow server IP address override at runtime** to be able to modify the IP address of the OS deployment server when starting up the target.
9. Plug your USB key into a machine running the Web interface extension and specify its address.
10. Choose the drive matching your USB key.
11. Click **Finish** to close the wizard.

Use the USB drive to boot the target.

Creating a kernel-based network boot CD or DVD with the wizard

1. Go to the **Task templates**, the **System Profiles**, or the **software modules** page.
2. Click **Generate media** at the bottom of the page.
3. Select **Create a network boot CD/DVD** and click **Next**.
4. Specify the operating system on which to boot the target. Select **Linux** to load a Linux deployment engine environment, **Windows** to load a WinPE deployment engine, or **Both** to have the two.

   Note:

   - If you use a network boot media to deploy Linux profiles, you cannot use the HTTP protocol. For this reason when you create a deployment scheme

    ensure you do not select the **Download files with a network share when applicable** option in the **Network settings** section.
  - If you use a network boot media and want to erase hard disk content, your media must contain a WinPE deployment engine. Therefore, you must select either **Windows** or **Both**.
5. If you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. If you want to obtain the target IP address through DHCP, select **Dynamic IP address with DHCP**, and click **Next**.

   If you want to use a fixed IP address for your target instead of having it go to the DHCP server, select **Static IP address**, and click **Next**.
   a. Enter the target IP address, gateway, and network mask.
   b. (*Optional*) Select **Allow IP address override at runtime** to be able to modify the target IP address when starting up the target.
   c. Click **Next**.
7. Enter the IP address of the OS deployment server.
8. (*Optional*) Select **Allow server IP address override at runtime** to be able to modify the IP address of the OS deployment server when starting up the target.

   Note: When you create the network boot CD or DVD in a multiserver infrastructure, ensure that the OS deployment servers share the same password and port number. The network boot CD or DVD works only if you specify the IP address of a OS deployment server having the same password and port number of the OS deployment server that generated the ISO file.
9. Click **here** to download the ISO file.
10. Click **Finish** to close the wizard.

The generated ISO file can be burned to create the network boot CD.

To start a target over the network using your OS deployment server without booting through PXE, start the target on the network boot CD and the target automatically connects to the OS deployment server.

Creating a kernel-based network boot USB drive with command lines

You can create a network boot USB drive, which Tivoli Provisioning Manager for Images can use when a target cannot boot from the network.

Install the rbagent, also known as web interface extension, on a Windows target. The USB drive must be formatted as FAT32 or NTFS. Existing files on the USB drive are not deleted. USB keys already filled with a bootable operating system might not work.

The command line must be used only when the web interface is either inappropriate or unavailable.

- If you want to obtain the target IP address through DHCP, use this command line:
  - Windows On Windows operating systems

```
rbagent.exe -s <OSD_server_ip_address>:<OSD_server_password>
rad-mkbootusb <drive>
<USB_OSD_server_ip_address> <USB_OSD_server_password>
[allowsrvipoverload] [nowpe|preferwpe]
[bootopt nnn] [clearcmos]
```

where:

OSD_server_ip_address
Is the IP address of the OS deployment server.

OSD_server_password
Is the password for the administrative user (typically admin) on your OS deployment server.

drive Is a drive letter of the Windows target where you run the rbagent command. The rad-mkbootusb command adds the requested files to the FAT32 or NTFS partition and makes it bootable. The drive must be already formatted. Existing files on the partition are not deleted.

USB_OSD_server_ip_address
Is the IP address of the OS deployment server that the target must contact, when it boots from the USB drive.

USB_OSD_server_password
Is the password of the OS deployment server that the target must contact, when it boots from the USB drive.

allowsrvipoverload
Allows you to choose an OS deployment server later, from the target.

nowpe|preferwpe
Defines if a Linux deployment engine environment or WinPE environment is loaded from the USB drive, when a target boots from this USB drive, without accessing the network. Only when Linux deployment engine or WinPE is running, does the target connect to the network and try to contact an OS deployment server. If you deploy only Linux, specify prefermcp to skip the WinPE deployment engine. You can specify preferwpe only if there is a WinPE deployment engine on the OS deployment server.

bootopt *nnn*
Allows you to specify additional flags before the boot.

clearcmos
Resets the CMOS alarm fields if they are in an invalid state.

For example:

```
> C:\TPMfOSd Files\global\http\agents\rbagent.exe
-s 10.10.10.10:abcd rad-mkbootusb C: 10.10.10.10 abcd
```

- If you want to use a fixed IP address for your target instead of having it go to the DHCP server, use this command line:
  – On Windows operating systems:

```
rbagent.exe -s <OSD_server_ip_address>:<OSD_server_password>
rad-mkbootusb <drive>
<USB_OSD_server_ip_address> <USB_OSD_server_password>
fixed [fixed_ip_address] [fixed_netmask] [fixed_gateway_ip_address]
[allowsrvipoverload] [nowpe|preferwpe]
[allowipoverload] [bootopt nnn] [clearcmos]
```

where:

OSD_server_ip_address
is the IP address of the OS deployment server.

OSD_server_password
is the password for the administrative user (typically admin) on your OS deployment server.

drive is a drive letter of the Windows target where you run the rbagent command. The rad-mkbootusb command adds the requested files to the FAT32 or NTFS partition and makes it bootable. The drive must be already formatted. Existing files on the partition are not deleted.

USB_OSD_server_ip_address
Is the IP address of the OS deployment server that the target must contact, when it boots from the USB drive.

USB_OSD_server_password
Is the password of the OS deployment server that the target must contact, when it boots from the USB drive.

fixed_ip_address
Is the static IP address of the target you boot using the USB drive.

fixed_netmask
Is the netmask of the target you boot using the USB drive.

fixed_gateway_ip_address
Is the IP address of the gateway that the target uses.

nowpe|preferwpe
Defines if a Linux deployment engine environment or WinPE is loaded from the USB drive, when a target boots from this USB drive, without accessing the network. Only when Linux deployment engine or WinPE is running, does the target connect to the network and try to contact an OS deployment server. If you deploy only Linux, specify nowpe to skip the WinPE software module. You can specify preferwpe only if there is a WinPE software module on the OS deployment server.

allowipoverload
Allows you to define IP settings manually on the target.

bootopt *nnn*
Allows you to specify additional flags before the boot.

clearcmos
Resets the CMOS alarm fields if they are in an invalid state.

You can now boot the target using the network boot USB drive instead of the network card. To use the PXE emulation USB key, insert the USB key into the drive and restart the target. If your machine does not boot from the USB key, check the BIOS boot list to see if your USB drive is included in the boot sequence and is listed before the hard disk. Most machines also allow you to select the temporary boot device without changing the boot sequence in BIOS.

Creating a kernel-based network boot CD or DVD with command lines

This mode must be used only when the web interface is either inappropriate or unavailable.

Note: When you create the network boot CD or DVD in a multiserver infrastructure, ensure that the OS deployment servers share the same password and port number. The network boot CD or DVD works only if you specify the IP address of a OS deployment server having the same password and port number of the OS deployment server that generated the ISO file.

- If you want to obtain the target IP address through DHCP, use these command lines:
  - UNIX Linux On UNIX and Linux operating systems

    ```
    #./rbagent -s <target_ip_address>:<target_password>
    rad-mkbootcd <full_path_to_boot_iso>
    <target_ip_address> <target_password>
    ```

  - Windows On Windows operating systems

    ```
    rbagent.exe -s <target_ip_address>:<target_password> rad-mkbootcd
    <full_path_to_boot_iso> <target_ip_address> <target_password>
    ```

  where:

  `target_ip_address`
  Is the IP address of the OS deployment server.

  `target_password`
  Is the password for the administrative user (typically `admin`) on your OS deployment server.

  `full_path_to_boot_iso`
  Is the full path to the `.iso` file you want to create on the target where you run the `rbagent` command.

  For example:

  ```
  > C:\TPMfOSd Files\global\http\agents\rbagent.exe
  -s 10.10.10.10:abcd rad-mkbootcd C:\boot.iso 10.10.10.10 abcd
  ```

  This creates a file called `boot.iso` in `c:\` which can be burned onto a CD.

- If you want to use a fixed IP address for your target instead of having it go to the DHCP server, use these command lines:
  - UNIX Linux On UNIX or Linux operating systems:

    ```
    #./rbagent -s <target_ip_address>:<target_password> rad-mkbootcd
    <full_path_to_boot_iso> <target_ip_address>
    <target_password> [fixed_ip_address]
    [fixed_netmask] [fixed_gateway_ip_address]
    ```

  - On Windows operating systems:

    ```
    > rbagent.exe -s <target_ip_address>:<target_password> rad-mkbootcd
    <full_path_to_boot_iso>
    <target_ip_address> <target_password>
    [fixed_ip_address] [fixed_netmask] [fixed_gateway_ip_address]
    ```

  where:

  `fixed_ip_address`
  Is the static IP address of the target you boot using the CD.

  `fixed_netmask`
  Is the netmask of the target you boot using the CD.

  `fixed_gateway_ip_address`
  Is the IP address of the gateway the target uses.

The generated ISO file can be burned to create the network boot CD.

To start a target over the network using your OS deployment server without booting through PXE, start the target on the network boot CD and the target automatically connects to the OS deployment server.

Creating a standard WinPE network boot CD or DVD in kernel-free mode with the wizard

Before you create the Windows PE-based network boot CD or DVD, ensure that you configured your WinPE deployment engines to match your target models and to contain the critical drivers for the specific target hardware.

Create this CD/DVD if you want to deploy Windows operating systems without using PXE, minimizing hardware compatibility issues.

1. Go to the **Task templates**, the **System Profiles**, or the **Software modules** page.
2. Click **Generate media** at the bottom of the page.
3. Select **Create a standard Windows PE-based media** and click **Next**.
4. Select both the deployment engine type and the architecture to use based on the operating system you want to install. If you choose the **Select the best available (Auto)** option and both 32-bitWinPE 3.x and WinPE 4.x are available, the system selects WinPE 4.x. You can override the default engine architecture and select the 64-bit architecture.
5. If you have more than one WinPE deployment engine, select the target models on which you want to use your media. The WinPE deployment engines matching the selected target models are included in the media.
6. Select the **Inject all available drivers** check box, if you are creating your media based on a model that is not known to Tivoli Provisioning Manager for OS Deployment. Using this option, all compatible drivers are added. Click **Next**.
7. To create a network boot CD/DVD, do not select any option on this panel and click **Next**.

   Important: Selecting any profile or software module takes you to the offline deployment media creation, which is described in "Creating a standard WinPE-based CD/DVD or USB for deployment in kernel-free mode" on page 105.
8. If you want your WinPE deployment engine to use a dynamic IP address through DHCP for your target during the provisioning, select **Dynamic IP address with DHCP**, and click **Next**.

   If you want your WinPE deployment engine to use a fixed IP address for your target instead of having it go to the DHCP server, select **Static IP address**, and click **Next**.
   a. Enter the target IP address, gateway, and network mask.
   b. (*Optional*) Select **Allow IP address override at runtime** to modify the target IP address when starting the target.
   c. Click **Next**.
9. Enter the IP address of the OS deployment server, or choose from a list of available addresses.
   a. Optionally select **Override OS deployment server address at run time** to modify the IP address of the OS deployment server when starting the target

Note: When you create the network boot CD or DVD in a multiserver infrastructure, ensure that the OS deployment servers share password and port number. The network boot CD or DVD works only if you specify the IP address of a OS deployment server having the same password and port number of the OS deployment server that generated the ISO file.

b. Optionally select **Automatically discover server on local subnet** to enable the server discovery feature. In this way, any server that belongs to the same local subnet can be discovered and contacted.

   Note: Ensure that the OS deployment servers of the same local subnet share the same password.

10. If your target startup sequence is first CD or DVD and second hard disk, make sure that **CD/DVD will boot at user request only** is selected to boot from the network boot CD/DVD only after user interaction. This is because the deployment flow must first start with a boot from the WinPE deployment engine available in the network boot CD/DVD while successive boots must be performed from the hard disk where the WinPE deployment engine was cached. For this reason, the first time that the target boots, you must press any key to boot from the CD/DVD and start the deployment. Successive unattended reboots fall back to hard disk until the deployment is completed.
11. Specify where the Web Interface extension (rbagent) needed to generate your ISO image is installed. Select the Web Interface on your local computer or specify the IP address of another computer in the network.
12. Specify the path and file name of the .ISO file that will contain the deployment engine.
13. After a few minutes the CD/DVD media is created. Click **Finish** to close the wizard. In the specified directory you can see the .ISO file.

The generated ISO file can be mounted on the target or burned to create the network boot CD.

To start a target over the network using your OS deployment server without booting through PXE, start the target on the network boot CD and the target automatically connects to the OS deployment server.

Using a network boot CD

When PXE network boot is not available in your network, use a network boot CD to start up your target.

From the OS deployment server create an ISO image of a network boot CD.

1. To boot virtual machines, mount the ISO image. To boot target computers, burn a CD/DVD from the ISO image and use it on your targets.
2. If you are using an original WinPE network boot CD during the first target boot, press any key to boot from the network boot CD and run the WinPE deployment engine.
3. The target connects to the network and contacts the OS deployment server. From the OS deployment server you can now submit any task on the targets. If you have enabled the server discovery feature, you can contact any server in the local subnet.

Booting on the network when the target is missing network drivers

Using Tivoli Provisioning Manager for Images you can boot on the network even if your target is missing network drivers and its model is unknown to the OS deployment server.

You are attempting to boot, through a network boot media, a new target that has a model unknown to the OS deployment server. Your target is missing network drivers and cannot therefore boot on the network as intended. Because the target model is unknown to the OS deployment server, you cannot bind the necessary drivers in the network boot media.

The solution is to import the PCI inventory of the target on the OS deployment server, to bind the needed drivers to the model, to recreate a network boot media, and to use this media to boot the target on the network successfully.

1. Export the PCI inventory of the target on a USB key or on a floppy disk.
   a. Create a network boot media.
   b. Boot your target with this media.
   c. If there are any missing drivers, and if you do not have a USB key already plugged in, the target waits until you have inserted a floppy disk or a USB key.
      1) If a file called `newhost.ini` already exists on the media, it is renamed.
      2) A file called `newhost.ini` is created on the media containing the PCI inventory of the target.
2. Import the PCI inventory of the target on the OS deployment server.
   a. Insert your media in the server.
   b. Go to **Server** > **OS deployment** > **Target Monitor**.
   c. Click **Import targets**.
   d. Type in the location of the `newhost.ini` file and its name, or browse to locate it, and click **OK**.
3. Re-create you network boot media
   a. If you have more than one WinPE deployment engine per architecture, check with which WinPE deployment engine your new target model matches.
   b. Bind the necessary network drivers to this WinPE deployment engine.
   c. Re-create a network boot media, selecting at least the WinPE deployment engine to which you have bound the drivers, and selecting **Optimized for Windows**.
4. Boot your target with the newly-created network boot media.

Because the target now has the appropriate drivers, it can connect to the network and contact the OS deployment server.

You can now use your target like any other target that is booted through a network boot media.

Chapter 11. Tools and additional features

This section provides information the disk content blanking feature and on the deprecated software snapshots.

Erasing hard disk content

Permanently erasing the content of a hard disk can prove necessary for confidentiality reasons.

To erase the content of a hard disk, you must have a WinPE deployment engine on your OS deployment server.

Note: If both WinPE 3.x and WinPE 4.x are available, WinPE 4.x is used. If target does not support WinPE 4.x, to run this task ensure that model patterns for WinPE 4.x do not match the target model.

For BIOS targets you need a 32-bit WinPE, while for UEFI targets you need a 64-bit WinPE. If you booted your target with a network boot media, the WinPE deployment engine must be present on the network boot media. In this case only BIOS targets are currently supported.

When a computer changes hands or purpose, you might have to make sure that the new user cannot recover data previously stored on the hard disk. To do so, the hard disk is completely written over with meaningless data, thus permanently erasing all previously stored data. The process can take up to a few hours, as every bit on the disk is written over.

Note: Erasing the content of the hard disk is a non reversible process which must be used with caution.

To erase the hard disk content of a target:

1. Go to the **Target Monitor** page
2. Select the targetor targets on which you want to erase the hard disk
3. In the contextual menu, select **Additional features**
4. In the **Additional feature wizard**, select **Destroy hard disk contents**
5. Follow the instructions of the wizard. It displays five disposal methods. Depending on the method you choose, you get a description of how the selected method erases the contents of the hard disk.

Performing wake-up or reboot operation on targets

The same deployment options are also available from the **Additional feature wizard** of the target monitor page.

Using the advanced features wizard from the target monitor page, select one or multiple targets and specify if you want to perform a wake-up or reboot operation.

1. Go to the **Target Monitor** page
2. Select the target or targets on which you want to perform the wake-up or reboot operation.
3. In the contextual menu, select **Additional features**

4. In the **Additional feature wizard**, select **Run Wake-up or reboot**. Follow the instructions of the wizard.
5. You can choose to:
   - Try to wake- up the targets currently switched off using WOL.
   - Try to wake- up the targets using the management interface.
   - Try to reboot the targets running the Web interface extension.
6. You can choose to:
   - Start the activity immediately.
   - Start the activity at scheduled time by specifying some scheduling options.
7. Click **Next** to start the activity.

Software snapshots

Using software snapshots is strongly discouraged. Current versions of the product can redeploy software snapshots which were created with older versions of the product. Software snapshot redeployment is supported for compatibility with earlier versions only. Creation of software snapshots is deprecated.

Limitation of the technology

Installing software by software snapshot is not intended as a general alternative for software installers. It is only safe when used in the correct environment, and when the software snapshots have been created carefully.

The difference between a real installer and a software snapshot is that whereas the installer can be aware of the present state of the computer and can act accordingly, the software snapshot is applied blindly and will therefore only do the correct work if the computer is in a similar state as the reference image on which the software snapshot was created.

The good point with the use of software snapshots is that as they are applied as part of a complete installation process, the environment is precisely known and does not depend on any previous user interaction or any past action performed on the target. The initial computer state is completely under control. It is therefore possible to safely use software snapshots.

The purpose of creating software snapshots is obviously to reuse them in several different circumstances, or to combine them in several ways. However, when doing these combinations, you must keep in mind that if some of the software snapshots are not fully independent one of the other (which is the ideal case, but which is not always possible), you must apply them in the correct order so as to reproduce the same environment originally present when each software snapshot was created.

Special care must be taken if software snapshots are used to handle hardware-related components. The binding of hardware drivers into the operating system can be tricky, and installing the same device in two different computer models can lead to very different registry keys, which can in some case make it impossible to use a common software snapshot.

When using software snapshots, you must also be aware of the fact that NTFS security attributes associated with files are also part of the software snapshot. However, the definition of users is typically not part of the software snapshot, but of the reference image. Therefore, you must avoid creating software snapshots with special user-related permissions, as it might lead to permissions problems if the user does not exist in the system profile being used for the final deployment.

Restoring software snapshots

You can only restore software snapshot which were created with old versions of the product. This feature is maintained only for backward compatibility.

Before restoring a software snapshot, you must ensure that you have an operating system properly installed on your hard disk. You might want to restore a system profile first.

1. Go to **Server** > **OS deployment** > **Target Monitor**.
2. Select the target on which you want to restore the software snapshot.
3. In the contextual menu, select **Additional features**.
4. Select **Restore a profile** and click **Next**.
5. Select the software snapshot and click **Next**.
6. Follow the remaining instructions of the wizard.

Binding menu

The binding menu is a task that can be scheduled on an idle target to show bound tasks, if any.

The administrator is responsible for defining the binding rules or explicit bindings on the target.

When the binding menu is created, either through a new bound or a PXE boot, it will not change until it is destroyed and recreated.

All the binding menus have an **update** button to recreate the menu with the latest changes.

Chapter 12. Task Customization Flow

Introducing the Task Customization Flow

This section provides a quick overview of the Task Customization Flow, also known as Toolkit V7, delivered with the Tivoli Provisioning Manager for OS Deployment 7.1.1.11.

The Tivoli Provisioning Manager for OS Deployment Toolkit V5 has been on the market for years and is positively adopted by customers interested in implementing Operating Systems deployment solutions tailored to their business needs. The simplicity is one of the major asset of the Toolkit and the main flow can be summarized as follows:

- The bare metal machine runs the network boot (PXE boot) connecting to the product server.
- The product pre-installation environment (Kernel) and the Start page are downloaded.
- The custom code is run in the Toolkit API layer.

Starting from this simple scenario, customers were able to implement OEM solutions for OS deployment or even a single deployment step in a larger business process.

Tivoli Provisioning Manager for OS Deployment solution is itself built upon the Toolkit technology: it is designed and implemented as an extension to the Toolkit API layer to provide easy-to-use deployment flow capabilities through advanced user interfaces and scripted activities.

The analysis of the Toolkit based implementations (including customers and IBM) highlighted significant common aspects and suggested the possibility by the customers to do the off-load of some code leveraging IBM built-in flows. Toolkit V7 allows to use such building blocks as built-in actions to compose custom tasks implementing the desired deployment logic.

The Tivoli Provisioning Manager for OS Deployment Toolkit V7 is designed to overcome the following issues present in the Toolkit V5:

- Provided an alternative to the kernel approach by leveraging light SysLinux bootloaders for BIOS machines.
- Maintain the customers scripts compatibility by offering the same execution environments but on the standard pre-installation environments (WinPE, Linux DE).
- Include the custom code into custom task to easily manage them and leverage the product infrastructure.
- Provide a library of built-in actions to easily compose Custom task re-using IBM code.

Requirements

The compatible operating systems and architectures for the Toolkit are a subset of the ones supported by Tivoli Provisioning Manager for OS Deployment, that is:

- Windows: x86-32 and x86-64 BIOS architectures (not UEFI architectures)
- Linux: x86-32 and x86-64 BIOS architectures (not UEFI architectures)

For more information, see Installation requirements

Technology

The kernel-free flow provides an alternative to the legacy kernel approach to run custom actions on the targets. It is based on SysLinux bootloaders (for BIOS) and it can chain-load a standard operating system environment to run deployment actions:

- Linux Deployment Engine (Linux DE): to start a minimal Linux environment using a RAM disk
- Windows PE (Win PE): to start a minimal Windows environment using a RAM disk
- Supported Windows and Linux OS installed on the hard-disk

The custom code can then leverage a larger set of supported batch scripting and programming languages within these environments.

Execution environment

Built-in actions are provided to change the task execution environment in one of the following:

- Windows PE
- Linux Deployment Engine
- Running Operating Systems: MS Windows and Linux OSs on Intel

Toolkit V7 implements a finite-state machine model to provide switching execution environment capabilities; given the current context in which the task is running, for example, the operating system, the built-in action is able to prepare the target to resume the task in the requested environment, for example, Windows PE.

Note: UEFI systems are not supported.

Main flows and its differences

You can work on the Tivoli Provisioning Manager for OS Deployment target both in online and offline mode.

To enable the target to work in offline mode, the target must work without the connectivity to the Tivoli Provisioning Manager for OS Deployment server and the cache containing the offline mode prerequisites must have been created on the target.

In the offline mode scenario, both a CLI and a user interface are available on the target.

In the online mode scenario:

- the target connects to the server and then performs the scheduled task. In offline mode, the target uses the cache resources to run the submitted task.
- the task progress status and control information are contained in the server file system or database. In offline mode, they are stored in the target cache.

- the resources used to accomplish the job are in the server file system. In offline mode, they are stored in the target cache.
- the required deployment engine is downloaded on demand from the server. In offline mode, the required deployment engine must be locally cached.
- the mechanism to maintain the control of the workstation configures the target to boot from the network and uses the Tivoli Provisioning Manager for OS Deployment server fallback MBR (F12 during the initial network boot). In offline mode, a temporary MBR is installed to restart the cache when running a task with transitions.

Set of commands

This section provides a detailed description of the Tivoli Provisioning Manager agent commands available with the Toolkit v7.

Create custom task template

Creates a custom task template to execute custom code against targets.

Command syntax

```
rad-createscheme  <ini-path> [update]
```

Parameters:

`ini-path`: the path to the INI file is absolute and defines the task template. For more details, see the sections “Available built-in actions to compose a custom task template” on page 374 and “Samples and use cases” on page 391.

`update`: if specified causes the update of an existing template in case it has been previously defined in the system.

Examples:

`rbagent.exe -d -v 4 rad-createscheme c:/mytask.ini`

The `mytask.ini` file has the following content:

```
[CUSTOM]
name=mytask
description=This is a sample task doing nothing
numactions=1
ON_FAILURE_REBOOT=false
password=XXXXX
timeout=20

[Action_0]
name=RunCustomScript

[Parameters_Action_0]
script=donothing.rbc
```

The section **`[CUSTOM]`** describes the task and its configuration:

`name`: The unique identifier of the task

`description`: The description of the file (displayed as a name in the UI)

numactions: The number of built-in actions that compose the task.

ON_FAILURE_REBOOT: The default value is true. If false will not cause the agent to reboot in case of task failure when running into the Operating system. If missing or set to true then a failure into the Operating system will result into a reboot of the same.

password: Creates a password entry using the **rad-hidepassword PASSWORD md5** command to encrypt.

timeout: Creates a time-out entry in the CUSTOM section of the custom task.

The section **[Action_xx]** represents the declaration of a built-in action to be executed by the task.

name: The name of a supported built-in action.

The section **[Parameters_Action_0]** represents the parameters of the action number xx: One by one all the parameters are listed by name and value.

Delete custom task template

Deletes a custom task template previously created.

Command syntax

```
rad-deletescheme  <Task=...|TaskName=...> [force]
```

Parameters to identify the task template:

Task: The task identifier.

TaskName: The task name.

Parameter to manage the deletion:

force: if specified cancel all pending tasks and then delete the template.

Examples:

```
rbagent.exe -d -v 4 -s <server ip>:pw rad-deletescheme Task=mytask
```

```
/rbagent.exe -d -v 4 rad-deletescheme TaskName= "This is a sample task
doing nothing"
```

Run custom task on targets

Runs the custom task on the targets to install the OS with a custom procedure implemented by the task template.

Command syntax

```
rad-scheduletask  <IP|MAC|SN|HostName|Description>=... <Task=...|TaskName=...> [reboot|pxeboot] [wak
```

Parameter to identify target host:

IP: IP address of the host.

MAC: mac-address of the host (the address must be in the following format: ABCDEFABCDEF).

SN: host serial number.

HostName: hostname of the host.

Description: description of the host.

Parameters to identify the task template:

Task: The task template identifier.

TaskName: The task template name.

Parameters to control host booting:

reboot: force host reboot.

pxeboot: force boot from the network.

wakeup: Turn on the target using WakeOnLan.

Examples:

```
rbagent.exe -d -v 4 rad-scheduletask Task=mytask IP=10.10.1.120
```

```
rbagent.exe -d -v 4 rad-scheduletask Task= mytask MAC=000C29AE9C3C
```

Run offline custom task on the target

Runs the custom task on the target in offline mode: the task template must exist on the local cache previously prepared.

Command syntax

```
rad-executetask <Task=...|TaskName=...>
```

Parameters to identify the task template:

Task: The task template identifier.

TaskName: The task template name.

Examples:

```
rbagent.exe -d -v 4 –e AGENT_WINSTALL –o rad-executetask Task= mytask
```

```
rbagent -d -v 4 –e AGENT_LINSTALL –o rad-executetask Task= mytask
```

List all deployment schemes

Lists all deployment schemes..

Command syntax

```
rad-schemelist  [details [Type=...]]
```

Examples:

```
rbagent.exe -d -v 4 -s <server ip>:pw rad-schemelist details Type=CUSTOM
```

```
rbagent -d -v 4 -s <server ip>:pw rad-schemelist
```

Available built-in actions to compose a custom task template

The following built-in actions are used in the INI file to compose a custom task template.

BootOnWinPE

Allows you to boot on WinPE to run custom tasks.

Parameters

- usePhysical [Optional, Default=false]: if true use URL like disk://0:1; if false use URL like local://root/c$

  When you run the built-in action BootOnWinPE, use the parameter usePhysical=true to have disk access with the product URL like "disk://0:0" on Windows PE.
- arch [Optional, Default=x86]: defines WinPE architecture to be started; allowed values are x86 or x86-64
- enginePath: if provided, defines the folder under <cachepath>/global/engines where the cached WinPE to be booted was previously stored. Otherwise network boot is assumed and Bios Fallback MBR might be used [Optional for the Online mode, Default assumes network boot, Required for the Offline mode]
- prepare_nextaction [Optional] :marks the action as completed.
- petype [Optional]: possible values are PE3 and PE4.

Examples

Invoked as a built-in action

```
[CUSTOM]
name=wpeboot
description=wpeboot
numactions=1
[Action_0]
name=BootOnWinPE
```

Invoked as a primitive within a custom action

```
struct {} bootwpeparams;
if (Offline()) bootwpeparams[(str)"enginePath"] := (var)"enginewpe";
BootOnWinPE(bootwpeparams);
```

BootOnLDE

Allows you to boot on Linux DE to run custom tasks.

Parameters:

- enginePath: if provided, defines the folder under <cachepath>/global/engines where the cached Linux DE to be booted was previously stored. Otherwise network boot is assumed and Bios Fallback MBR might be used [Optional for the Online mode, Default assumes network boot, Required for the Offline mode]

- `prepare_nextaction` [Optional]: marks the action as completed.

Examples

Invoked as a built-in action

```
[CUSTOM]
name=ldeboot
description=ldeboot
numactions=1

[Action_0]
name=BootOnLDE
```

Invoked as a primitive within a custom action

```
struct {} params;
params[(str)"enginePath"] := (var)"enginelde";
BootOnLDE(params);
```

BootOnHD

Allows you to boot on HD.

Parameters

- `diskno` [Optional, Default=0]: disk number (as recognized by BIOS) on which boot on
- `partno` [Optional, Default=0]: partition number (as described by Partition Table) on which boot on
- `prepare_nextaction` [Optional]:marks the action as completed.

Examples

Invoked as a built-in action

```
[CUSTOM]
name=hdboot
description=hdboot
numactions=1


[Action_0]
name=BootOnHD

[Parameters_Action_0]

diskno=0
partno=1
```

Invoked as a primitive within a custom action

```
struct {} params;
params[(str)"diskno"] := (var)0;
params[(str)"partno"] := (var)1;
BootOnHD(params);
```

RebootOnEnv

Allows you to reboot the system and allows different set of parameters to identify the target booting environment.

Simple reboot parameters

- target_env "UNKNOWN": defines no target for the reboot. A simple restart occurs.

Target parameters

- target_env:The target environment for the reboot. Parameters related to the BootOnWinPE , BootOnLDE, or BootOnHD actions can be passed.

To boot on WPE, the value must be *AGENT_WPE*.

To boot on Linux DE, the value must be *AGENT_LINUXDE*.

To boot on generic operating system, the value must be *AGENT_LINSTALL* or *AGENT_WINSTALL* Parameters related to the BootOnHD action has to be passed.

Example

Invoked as a primitive within a custom action

```
struct {} params;

params[(str)"enginePath"] := (var)"enginelde";
params[(str) "target_env"] :=(var)'AGENT_LINUXDE';
RebootOnEnv(params);
```

RunCustomScript

Allows you to run Toolkit V5 RemboC scripts. This feature acts as a wrapper to keep legacy compatibility with Toolkit V5 based applications.

Parameters

script [required]: Name of the file .rbc, including the path of the file relative to <DataDir>\global\remboc, where <DataDir> is the product directory for internal files. For more details on <DataDir>, refer to the documentation in the IBM Infocenter: Global parameters.

Example

Invoked as a built-in action

```
[CUSTOM]
name=myscript
description=myscript
numactions=1

[Action_0]
name=RunCustomScript

[Parameters_Action_0]
script=test.rbc
```

WriteCache

Allows you to create the Tivoli Provisioning Manager for OS Deployment cache, on the provided disk or partition, to work in offline mode. The partition maintaining the cache must always be FAT32. If the FAT32 cache partition is created from a Linux OS with mkfs.vat command, make sure that the HiddenSectors parameter is set by using the option -h.

Parameters:

- target: [mandatory] The cache partition mount point. Note that the cache must be created as root folder. For example:

  disk://0:1/ or local://root/e$
- loader: [optional] The type of loader to be used when working with cache, admitted values are:
  - RB: The Kernel loader.
  - KF: The Kernelfree loader.

Example

Invoked as a primitive within a custom action

```
struct {} params;
params[(str)"target"] = (var)"disk://0:1/";
params[(str)"loader"] = (var)"RB";
WriteCache(params);
```

Note: When running WriteCache with the Kernelfree loader, a Syslinux bootloader is left on the target machine to enable future OS deployment. In case you transfer the hardware, you should comply with the GPL license. You can also use the DeleteCache built-in action to remove the Syslinux bootloader.

CacheWPE

Allows you to save into the cache the engine to install Windows. In this way, it is possible to start the engine from the disk.

Parameters:

- target: the folder where the WPE engine will be saved. This folder will be used with BootOnWinPE to start the local engine. For example, if the parameter target is given the value mydir, the WinPE is saved into the following cache path:

  local://root/cachepartition/.TPMfOSd/global/engines/mydir
- type: if provided, defines how to select the WinPE engine to be cached [Optional, Default = SRVBYMODEL]. Possible values are "SRVBYMODEL", "SRVBYID".
- arch: force cached WinPE to be x86 or x86-64 [Optional, Default is x86]
- engineId: required if type=SRVBYID. Defines Deployment Engine ID to be stored
- srvip: if provided, cached WinPE will boot online. Otherwise Winpe will be started offline [Optional, Default asumes offline]
- srvpass: requested if srvip provided, required if srvip provided
- clnip, clnmask, clngateway: if provided (all of them must be set) set fixed IP on cached WinPe once started. Otherwise DHCP is assumed.
- petype: the possible values are PE3 and PE4.

Example

Invoked as a primitive within a custom action

```
struct {} wpeparams;
wpeparams[(str)"target"] = (var)"enginewpe";
CacheWPE(wpeparams);
```

ActivateCache

Allows you to activate (deactivate) the cache to enable (disable) the offline mode.

The built-in action cannot be used for the kernel free if the WinPE started in raw mode, that is, if the BootOnWPE was used with the parameter usePhysical = true.

When you activate the cache:

- the original Master Boot Record (MBR) is backed up and an internal persistent MBR is stored to start the cache
- at every reboot the persistent MBR is kept to preserve boot on the cache
- control is grabbed by the cache starting the custom binding menu if previously created with BindCustomTasks builtin action, or performing any pending action, if previously started, for example BootOnWinPE.

When you deactivate the cache:

- backed up MBR is restored
- restoration preserves current disk signature and current partition table, the original bootable partition and binary code (0->440 bytes) are restored from the backup

Parameters:

- activate: true or false to enable the offline mode.

Example

Invoked as a primitive within a custom action

```
struct {} params;
params[(str)"activate"] = (var)"true"; // false to deactivate
ActivateCache(params);
```

DeleteCache

Allows you to delete the cache previously created. Removes the .TPMfOSd folder, REMBOLDR, the extengine.ini, and the boot files if explicitly requested. Use this action only after you have deactivated the cache as described in section "ActivateCache."

Note:

- You must be online to perform this built- in action.
- files (.TPMfOSd folder + REMBOLDR + extengine.ini + boot files) are deleted from the cache partition. Be aware when running this action.

Behavior:

- If no cache is found, exit
- if the cache has not been created with WriteCache, force=true parameter is needed
- boot files that might interfere with the normal boot process (bootmgr, boot folder) are removed only if removeBootFiles=true

Parameters:

- force: [Optional] if true removes the cache even if not persistent, meaning it was not created with WriteCache.
- removeBootFiles: [Optional] if true removes the bootable files used by the engines. By default, they are not removed not to interfere with the machine boot process. The files are:
  - bootmgr file
  - Boot folder

Example

Invoked as a primitive within a custom action

```
struct {} deleteparams;
DeleteCache(deleteparams);
```

BindCustomTasks

Allows you to create the binding menu for the custom tasks (online and offline modes).

Note: You can set time-out and password for the custom tasks bound to the menu.

- timeout: you create a time-out entry in the CUSTOM section of the custom task
- password: you create a password entry in the CUSTOM section of the custom task using the rad-hidepassword PASSWORD md5 command to encrypt such as in the following example:

  ```
  [CUSTOM]
  name=BindCustomTasks
  description=password
  numactions=1
  password=5F4DCC3B5AA765D61D8327DEB882CF99
  ```

  Maximum one WinPE is allowed but you can bind several custom tasks to it.

Behavior: the binding rules are created into the database (local or server) for the custom tasks. A menu with linked tasks is displayed at every system startup.

Parameters:

- tasks: comma "," separated custom task name to bound to the target
- engines: [Optional] comma "," separated cached engines name. This parameter is needed for the offline Kernel Free flow to specify the task execution environment defined in the tasks parameter:
  - hdboot_X_Y to boot on disk X and partition Y associated to bound tasks (replace X with disk number and Y with partition number). For example if disk name is 0 and partition name is 1, then you have hdboot_0_1.
  - the folder where WinPE or LinuxDE have been previously cached using CacheWinPE or CacheLDE commands, respectively. This folder is used with BootOnWinPE or BootOnLDE to start the local engine when running the selected task. For example, if the folder target is called mydir, the WinPE or LinuxDE to be run are started from the following cache path: local://root/cachepartition/.TPMfOSd/global/engines/mydir. If you specify:
    - tasks = task1, task 2
    - engines = folder1, folder2

task1 is run from environment in folder1 and task2 from folder2. For this reason, the number of tasks must correspond to the number of available engines.

Considerations for the boot on Hd task::

From the custom task menu, you can boot on OS by adding a custom task running the built-in action boot on hd.

Examples

Invoked as a primitive within a custom action

```
/*CREATE MENU ONLINE*/
struct {} menuparams;
menuparams[(str)"tasks"] = (var)"usertest, hdboot";
BindCustomTasks(menuparams);
```

```
/*CREATE MENU OFFLINE*/
menuparams[(str)"offline"] = (var)"true";
menuparams[(str)"engines"] = (var)"enginewpe, hdboot_0_1";
menuparams[(str)"tasks"] = (var)"usertest, hdboot";
BindCustomTasks(menuparams);
```

Note: The binding menu task is created only if the agent/target is started as IDLE; it is not scheduled automatically if a previous task is completed and a green/red banner is shown.

RunCommand

This action allows the execution of system command, a command executable or a batch script.

Parameters:

- `command`: the command to run. It can be a script with the extension `.sh` on Linux (it is run as `sh script.sh`), or `.bat` on Windows, or a system command.
- `parameters`: [Optional] use to specify the command parameters that will be provided as arguments on command invocation.
- `origin`: [Optional, Default=Local] defines where the command is located. Allowed values are `Local` or `Server`. If `Server` is indicated the command has to be located in the `global/remboc` directory on the server.
- `timeout`: [Optional, Default=288000] the maximum amount of time to wait for command to complete in cents of seconds. Default is 48 minutes.
- `onerror`: [Optional, Default=Stop] defines the action behavior in case of command error. Allowed values are `Stop` or `Continue`. If `Stop` is specified, a command error will cause the action to fail.
- `prefix`: [Optional Default=`local://root/x$/` on WinPE, `local://root/root/` on LinuxDE, `local://root/c$/` on Windows OS, `local://root/tmp/` on Linux OS] defines the root location on the target system and has to be a local URL. The parameter value represents the directory where the command is run.

  On Linux, if command parameter is `ls`, then on the target the command `ls` will be run:

  ```
  command=ls -> running "ls"
  ```

  If command parameter is `script.sh`, then on the target the command `sh script.sh` will be run:

  ```
  command=script.sh -> running "sh script.sh"
  ```

Examples

```
[CUSTOM]
name=MyCommand
description=TestMyCommands
numactions=1
ON_FAILURE_REBOOT=false
[Action_0]
name=RunCommand
[Parameters_Action_0]
command=CustomRun.bat
parameters = clean
origin = Server
timeout = 34000
onerror = Continue
prefix = "local://root/x$/"
parameters = clean
```

This built-in action is available as primitive in `remboc` scripts.

CacheLDE

This action stores locally the best LDE available for the target (eventually using the specified binding grid).

Parameters:

- `target`: the folder where the WinPE engine will be saved. This folder will be used with BootOnWinPE to start the local engine. For example, if the folder `target` is called `mydir`, the WinPE is saved into the following cache path: `local://root/cachepartition/.TPMfOSd/global/engines/mydir`
- `type`: if provided, defines how to select the LDE engine to be cached [Optional, Default = SRVBYMODEL]. Possible values are "SRVBYMODEL" (no other params needed), "LOCALDIR"(needed bootfile and ramfile);
- `bootfile`, `ramfile`: required if type=LOCALDIR. Defines the kernel image (krn) and the ramdisk image (.img) needed to boot LDE. In the same target path the `rbagent.linux` is needed.
- `srvip`: [Optional, Default assumes offline] if provided, cached LDE will boot online. Otherwise WinPE will be started offline
- `srvpass`: requested if `srvip` provided.
- `clnip`,`clnmask`,`clngateway`: if provided (all of them must be set) set fixed IP on cached WinPE once started. Otherwise DHCP is assumed.
- `clnmac`: [Optional] if provided, force to use a given MAC address to connect to server

Examples

Invoked as a primitive within a custom action

```
struct {} ldeparams;


ldeparams[(str)"target"] = (var)"enginelde";


CacheLDE(ldeparams);
```

Note: A Linux deployment engine comprising open source code under GPL license is left on the target machine to enable future OS deployment. In case you transfer the hardware, you should comply with the GPL and other open source licenses. You can also use the `DeleteCache` built-in action to remove the Linux deployment engine.

Built-in actions in custom Rembo-C API

The Toolkit version 7 provides custom tasks to execute custom code on the targets. The built-in actions can be either grouped into a custom task template or invoked as functions from the custom code. This section describes the syntax of the built-in actions as functions.

Every built-in action as function accepts a generic structure of parameters:

```
(var parameters)
```

which consists in a map having as its keys the parameters accepted by the built-in action.

For example, the built-in action CacheWPE can be invoked as function using its parameter “target":

```
struct {} wpeparams;
wpeparams[(str)"target"] = (var)"enginewpe";
CacheWPE(wpeparams);
```

The list of built-in actions available in RemboC scripts is as follows:

- `void BootOnHD    (var parameters)`
- `void BootOnWinPE (var parameters)`
- `void BootOnLDE    (var parameters)`
- `void RebootOnEnv (var parameters)`
- `void WriteCache  (var parameters)`
- `void CacheWPE    (var parameters)`
- `void ActivateCache (var parameters)`
- `void DeleteCache  (var parameters)`
- `void BindCustomTasks (var parameters)`
- `void CacheLDE (var parameters)`
- `void RunCommand (var parameters)`

Refer to “Samples and use cases” on page 391 for details about the custom script samples.

Refer to “Available built-in actions to compose a custom task template” on page 374 for details about the list of parameters supported for each built-in action.

General-purpose functions in custom Rembo-C scripts

The Toolkit V7 provides general-purpose functions that can be run from custom Rembo-C scripts. This section describes the usage of some of these functions.

Managing task configuration data

Task configuration data can be permanently stored into text files, according to the INI file format used by the Tivoli Provisioning Manager for OS Deployment product. In general, each task is associated with INI files where activity relevant information are stored in. In the same way, custom Rembo-C scripts can have their own information stored into these INI files.

The INI file format is a list of key-value elements grouped into sections with the following structure:

```
[Section_name]
key=value
[Section_name]
key=value
```

For example:

```
[Image_Capture]
Sysprep=OOBE
```

To manage task configuration data, the following public functions are available:

- **void setCustomActivityIniValue(str section, str key, str value)**

Stores the *key=value* pair under *section* for the target activity INI file. For example, to create the following section into the target activity INI file:

```
[Status]
step1="done"
```

run the following command from a custom Rembo-C script:

```
setCustomActivityIniValue("Status","step1","done");
```

- **var getCustomActivityIniValue(str section, str key)**

Retrieves the *key=value pair* under *section* from the target activity INI file. For example, to assign the string “done” to the variable value, run the following command from a custom Rembo-C script:

```
str value = getCustomActivityIniValue("Status","step1");
```

- **void setCustomActivityIniStruct(str section, var value)**

Stores each field of the variable *value* into *section* for the target activity INI file. For example, to create the following section into the target activity INI file:

```
[Status]
cpu="10"
mem="1"
```

run the following command from a custom Rembo-C script:

```
struct {
      str cpu = "10";
      str mem = "1";
} s_targetStatus;

struct s_targetStatus targetStatus;
setCustomActivityIniStruct("Status",targetStatus);
```

- **var getCustomActivityIniStruct(str section)**

Retrieves all the fields of *section* returned as a `struct` variable from the target activity INI file. For example, to return the following variable:

```
{ cpu: "10",   mem: "1" }
```

run the following command from a custom Rembo-C script:

```
getCustomActivityIniStruct("Status")
```

Completing task during environment transition

- **void prepareNextAction()**

It can be started from a custom RemboC script to mark the current custom action as completed and move to the next one (or complete the activity if no more actions are present in the plan).

For more details, see the section "Completing task during environment transition."

Working in offline and online modes

The following aspects must be considered when working in offline mode and switching mode between the online and the offline.

Custom tasks can be executed with or without server connectivity.

During the online mode, the server database is used as task data source. In the offline mode, a local database is stored in the local cache and used as task data source.

Even if some actions are required to run in online mode (WriteCache action) most of the custom tasks can also be scheduled and performed without server connectivity requiring the possibility to switch mode between online and offline.

Preparing the target to work in offline mode

The target must be enabled to work in offline mode. The typical preparation flow is as follows:

1. **Prepare the cache partition**

   Using the Toolkit APIs you can generate a partition on the disk to host the cache.
2. **WriteCache**

   Achieved by means of a library function (built-in action) preparing the cache to enable the offline mode.
3. **CacheWPE**

   Allows you to store the more appropriate WinPE deployment engine for the target to use it when working in offline mode.
4. **BindCustomTasks**

   Creates a menu which displays a list of tasks to the target user.
5. **ActivateCache**

   Installs the Tivoli Provisioning Manager for OS Deployment permanent MBR to make the cache as the default booting partition. In this way, Tivoli Provisioning Manager for OS Deployment takes control of the target.
6. **Reboot**

   Restarts the target and boots into the TPM for OS Deployment installed menu.

Switching modes

The scope of this feature is to mark the current task being executed as completed, and to switch execution flow from online to offline mode.

How to switch mode during custom actions

When running a custom action, start the public method `prepareNextAction()` from a custom rbc script to mark the current custom action as completed and move to the next one (or complete the activity if no more actions are present in the plan).

Note:

- An exception can be raised on errors.
- Use this method just before the primitives that will close your operational flow like `shutdown/reboot` and after all your operations have been completed. A running custom action can have unexpected results if completed in the middle of the execution.

How to switch mode during BootOnX actions

For the `BootOnHD`, `BootOnWinPE` and `BootOnLDE` built-in actions, use the parameter `prepare_nextaction` to prepare the task for the next action (or complete it if it is the last one).

Use this parameter when no agent runs after booting the target in the target desired environment.

The scope of this parameter is to mark the task as completed, during the last built-in actions, after `BootOnHD`, `BootOnWinPE` and `BootOnLDE` and to switch from online to offline mode.

Examples: How to switch from offline to online

When switching from offline to online, the pending tasks in the local database of the target, if any, are automatically set to complete.

1. The agent running a custom action in offline mode runs the following script:

   ```
   prepareNextAction();
   WriteFallbackMBR(!!"Redirecting boot process to next device\r\n"
   "(remove floppy or CDROM if you want to boot on the
   network)\r\n");
   Reboot();
   ```

   Result: The local database has no pending tasks, the target is forced to boot on the network and the server database is used.
2. The agent with a cached WinPE engine at path `enginewpe` configured to start online:

   ```
   struct {} wpeparams;
   wpeparams[(str)"target"] = (var)"enginewpe";
   wpeparams[(str)"srvip"] := (var) "10.10.0.5";
   wpeparams[(str)"srvpass"] := (var) "xxxxxxxxxxxxxxxxxxxxxxxx";
   CacheWPE(wpeparams);
   ```

   then running a BootOnWinPE action in offline mode with the following parameters:

   ```
   [Action_0]
   name=BootOnWinPE
   [Parameters_Action_0]
   enginePath=enginewpe
   prepare_nextaction=true
   ```

   Result: The local database has no pending tasks, the target is forced to boot on the cached WinPE engine that connects to the server.

3. The agent created the offline custom task menu with the following parameters:

   ```
   struct {} menuparams;
   menuparams[(str)"offline"] = (var)"true";
   menuparams[(str)"tasks"] = (var)"hdboot";
   BindCustomTasks(menuparams);
   struct {} params;
   params[(str)"activate"] = (var)"true";
   ActivateCache(params);
   ```

   where `hdboot` task is the following requiring `prepare_nextaction`

   ```
   [CUSTOM]
   name=hdboot
   [Action_0]
   name=BootOnHD
   [Parameters_Action_0]
   diskno=0
   partno=1
   prepare_nextaction="true"
   ```

   then booting from the hard disk and selecting `hdboot` from the offline menu.

   Result: The local database has no pending tasks, the target boots on the disk without any agent required to start on OS.

Examples: How to switch from online to offline

1. The agent running an online task from OS (`env=AGENT_WINSTALL`) is creating and activating the cache with the offline menu and the offline WinPE engine:

   ```
   struct {} params;
   params[(str)"target"] = (var)"disk://0:1/";
   params[(str)"loader"] = (var)"RB";
   WriteCache(params);
   struct {} wpeparams;
   wpeparams[(str)"target"] = (var)"enginewpe";
   CacheWPE(wpeparams);
   Printf("CUSTOM TASK MENU \r\n");
   struct {} menuparams;
   menuparams[(str)"offline"] = (var)"true";
   menuparams[(str)"tasks"] = (var)"usertest, wpeboot, reboot, hdboot";
   BindCustomTasks(menuparams);
   struct {} params;
   params[(str)"activate"] = (var)"true";
   ActivateCache(menuparams);
   struct {} bootparams;
   bootparams[(str)"diskno"] := (var)0;
   bootparams[(str)"partno"] := (var)2;
   bootparams[(str)"prepare_nextaction"] := (var)"true";
   BootOnHD(bootparams);
   ```

   The server database has no pending tasks, if the target is booted on the disk, the cache partition starts and the offline menu is displayed. If the target starts from the network, the server database is used.

2. The agent running an online task that will BootOnWinPE online, runs a custom script then a BootOnHD, marking the task as completed:

   ```
   [CUSTOM]
   name=wpeboot
   23
   [Action_0]
   name=BootOnWinPE
   ```

   ... any custom action in the middle...

   ```
   [CUSTOM]
   name=hdboot
   [Action_0]
   name=BootOnHD
   ```

```
[Parameters_Action_0]
diskno=0
partno=1
prepare_nextaction="true"
```

The server database sets the task as completed and the target starts from OS without any pending task.

Base flow customization

The Tivoli Provisioning Manager for OS Deployment infrastructure implements a set of "base flows" to manage the more relevant "interaction points" with the target during the deployment tasks. Operations performed during an operating system provisioning (such as target boot, tasks execution, file transfer, and so on) are "interaction points" managed by Tivoli Provisioning Manager for OS Deployment with predefined execution patterns (the "base flows") to support all the different use cases with a standard behavior.

More flexibility was introduced into the infrastructure to allow you to plug your own code and business logic to modify the behavior of the base flows according to your own requirements.

With respect to the Tivoli Provisioning Manager for OS Deployment Toolkit Version 5, you can provide specific functions called by Tivoli Provisioning Manager for OS Deployment during the execution of the base flows. These functions are invoked on a specified time only and are called by Tivoli Provisioning Manager for OS Deployment during predefined execution times. These functions are called only for the following actions:

- PXE boot redirection
- Default task execution
- Deployment engine for default task execution

Base flow technology

A new plug-in technology supports the base flow customization. Base flow customization plug-ins are loaded at Tivoli Provisioning Manager for OS Deployment server startup and are executed within the server and the agent context. Depending on the technology used at the target PXE-boot time, the custom plug-in will be invoked differently:

- If the target boots in kernel mode, the custom plug-in is loaded at the server startup but executed on the target.
- If the target boots in kernel-free mode, the custom plug-in is loaded and executed on the server during the target PXE-boot.

The following behaviors can be customized by the custom plug-in:

- PXE boot redirection: at target PXE-boot time, the custom plug-in can force the PXE-boot on a secondary server.
- Default task execution: at target PXE-boot time, the custom plug-in can force the execution of a default task (instead of the binding menu created on the targets without pending tasks).
- Deployment engine for default task execution: at target PXE-boot time, the custom plug-in can force to load WinPE or Linux DE (only for kernel-free targets).

Developing base flows

Kernel-free plug-in

The Kernel-free (KF) base flow plug-in allows you to configure:

- an alternate server to execute the PXE boot of a specific target
- the default task to be executed on the booting target
- the deployment engine to be started with the default task

The samplePluginKF.rbc file provided below is an example of kernel-free plug-in executing the redirection on an alternate server for targets with a specific model family and executing a default task for all the others. This plugin is executed on the server.

Sample:

```
/**
* plug-in function invoked at each PXE boot.
*/
var extension_function(var params)
{
  // two main parameters are passed to identify the booting target:
  //MAC: The macaddress of the machine
  //Model: The model of the machine.
  //The answer is made of an ini file including the calculated parameters.
  var myini = NewIniFile();
  if(StrMatchCI(params.Model,"IBM-90*"))
  {
     str resultingIP= "10.10.1.121";
     25
     SetIniValue(myini, "SERVER", "ip", resultingIP);
  }e
  lse
  {
     //Assign the default task (by ID) to be executed by the target when
     idle!!
     str defaultTask = "inventory";
     SetIniValue(myini, "TASK", "name", defaultTask);
     str resultingEngine = "AGENT_WPE";
     SetIniValue(myini, "ENGINE", "type", resultingEngine);
  }
  return myini;
}
//Register the extension function to manage the targets (clients) kernel free
PXE boot.
bool val = registerExtensionPoint('TARGET_BOOT','prova',extension_function);
```

The developed rbc/rbx file containing the custom business logic must be saved under the following predefined directory on the server file system:

```
global/serverext/SamplePluginKF.rbc
```

Notice the /serverext/ path since the plug-in runs on the server context.

Kernel plug-in

The Kernel base flow plug-in allows you to configure:

- an alternate server to execute the PXE boot of a specific target
- the default task to be executed on the booting target

The samplePluginRB.rbc file provided below is an example of kernel plug-in executing the redirection on an alternate server for targets with a specific model family and executing a default task for all the others. This plugin is executed on the agent.

Sample:

```
/**
* plug-in function invoked at each PXE boot.
*/
var extension_function(var params){
// two main parameters are passed to identify the booting target:
//MAC: The macaddress of the machine
//Model: The model of the machine.
//The answer is made of an ini file including the calculated parameters.
var myini = NewIniFile();
if(StrMatchCI(params.Model,"IBM-90*")){
str resultingIP= "10.10.1.121";
SetIniValue(myini, "SERVER", "ip", resultingIP);
}
26
else{
//Assign the default task (by ID) to be executed by the target when
idle!!
str defaultTask = "inventory";
SetIniValue(myini, "TASK", "name", defaultTask);
}
return myini;
}
//REgister the extension function to manage the targets (clients) kernel free
PXE boot.
bool val =
registerExtensionPoint('TARGET_BOOT_KN','prova',extension_function);
```

The developed rbc/rbx file containing the custom business logic must be saved under the following predefined directory on the server file system:

```
global/agentext/SamplePluginRB.rbc
```

Notice the /agentext/ path since the plug-in runs on the agent context.

Development guidelines and general considerations

- When working with the Toolkit in offline mode, the correct way to encrypt the password for a custom task is as follows:

  ```
  rad-hidepassword <pwd> md5
  ```

  While running the command:

  ```
  rad-hidepassword <pwd>
  ```

  Generates the wrong password encryption.
- When working with the toolkit in offline mode on some Windows platforms where BCD is missing such as Windows XP, when caching the WinPE engines without the Windows bcdedit.exe support, it is recommended to cache the WinPE engines using the TPM for OS Deployment engines where BCD is already preconfigured at creation time and patched as required by the CacheWPE built-in action.
- Except for BootOnWinPE, BootOnHD, and BootOnLDE, the default environment where the remaining built-in actions are executed is WinPE. This means that a kernel-free target, pxebooting and executing a WriteCache built-in action, will start in WinPE and then execute the WriteCache built-in action. To change this

behavior, it is suggested to enter the correct environment before running the `BootOnWinPE`, `BootOnHD`, and `BootOnLDE` built-in actions.

- Some Rembo script APIs have been modified and deprecated in favor of new ones.
- The deployment engines (WinPE/Linux DE) are now required and managed by the product: they provide two new execution environments: Windows PE and Linux Deployment Engine. Linux DE is automatically provided by Tivoli Provisioning Manager for OS Deployment, while for more information on how to work with Windows PE, see "Overview of WinPE deployment engines" on page 17.
- The preferred execution flow is to rely on the kernel free flow where the custom actions are run inside the execution environment provided by the deployment engines (WinPE, Linux DE).
- When running online (-s or PXE in kernel env) the cache is deactivated automatically meaning that the backed up MBR is replaced, and the task data source is on the server. After running online some actions (-s or PXE in kernel environment), the code to reactivate the cache must be started.
- To manage your targets, the Toolkit option is no longer available from the web UI. You can now use the option OS Deployment > Target Monitor.

Hints and tips to consider while writing Rembo-C scripts

- To leverage the built-in action capabilities for switching between execution environments, when you run the product agent binary (`rbagent`) from an operating system, for example after a boot on HD action, depending on the operating system you must add one of the following options to the command line:
  - `-e AGENT_WINSTALL`, for Windows
  - `-e AGENT_LINSTALL`, for Linux

  With this information, the Toolkit can detect the current state of the execution of the custom script and complete the scheduled action.

  Note: The `-e` option cannot be used at the end of the command.
- To show the console output on the target running a custom Rembo-C script, use the following functions:
- `ShowConsole()` and `HideConsole()`

  `ShowWindow()` - works only in rembokernel environment
- To show the splash skin when the rembokernel starts, use the following function:

  `LoadSkin()` - works only in rembokernel environment

  If this function is used outside the supported environment, it generates an exception that must be handled with an exception handler.
- When a custom Rembo-C script is waiting for user input on a UI display, make sure the script does not terminate, otherwise the custom action is considered as completed and the next action in the activity plan is started. To avoid this, at the end of the custom script you can add the following line:

  ```
  while (sem) { delay(sec_sleep*100); }
  ```

  where `sem` is a flag set to `false` when a UI callback functions is selected and the current action needs to be terminated. Callback UI functions will terminate the script upon a user action and the next action will be executed. When running Rembo-C functions that will reboot/shutdown a machine, the above script is skipped automatically.

- When using the kernel-free flow, do not modify the “startpage” column of the SrvHosts table in the internal product database: this field is used by the product to handle targets.

Samples and use cases

The following sections provide a quick hands-on guide to help you create your own *Task Customization* flow from scratch or from your existing *RemboC* scripts.

You can also find an example of task INI file and scripts with built-in actions used as primitive.

Preparation steps

Preliminary steps are needed to use the Task Customization flow:

- Tivoli Provisioning Manager for OS Deployment server must be installed and configured properly
- Required Deployment Engines must be created
- Custom Tasks must be prepared

The next sections will guide you through this process across some use-cases.

Install Tivoli Provisioning Manager for OS Deployment server

Installation process can be performed unattended (with a response file) or with user interaction: a few input parameters and minimal requirements are needed to install the Tivoli Provisioning Manager for OS Deployment server.

For more information, see Installation Guide.

Configure Tivoli Provisioning Manager for OS Deployment server

Setting kernel-free mode for unknown targets

After the installation, Tivoli Provisioning Manager for OS Deployment server can be configured to work in kernel-free mode (recommended mode) with all new targets running the following command:

./rbagent -d -v 4 -s 127.0.0.1:xxxxx rad-updateidlescheme DefaultOptions="KernelFree"

or by providing the requested parameter (kernel-free mode) within a custom configuration file (*custom.conf*) that will be appended to the *rembo.conf* configuration file at server installation (*CONFADDON* parameter).

The kernel-free parameter to be provided in the custom configuration file is described by the following line:

DefaultOptions KernelFree

This line must be provided in a text file used at installation time to be merged with default server installation settings. The installation parameter (for the unattended setup) requesting the installer to use a custom configuration file is:

CONFADDON="<path to custom.conf>"

where *custom.conf* is the custom configuration file containing the parameters that will be appended to the *rembo.conf* configuration file.

Setting kernel-free mode for known targets

The kernel-free boot parameter can also be changed on all known targets already existing in the Tivoli Provisioning Manager for OS Deployment database (if upgrading from previous versions).

To make known targets PXE-boot in kernel-free mode the command line to use is:

/rbagent -d -v 4 -s 127.0.0.1:xxxxx rad-registerhost IP=172.16.181.101 RemboOptions=65536

or the server-side scripts need to run a target registration with the *startpage* parameter:

host["startpage"] = (var) KERNELFREE_STARTPAGE; // KERNELFREE_STARTPAGE is a global constant

RegisterHost(host);

The target boot configuration can also be modified from the *Tivoli Provisioning Manager for OS Deployment Web UI → Target Monitor → Target Boot Settings.*

Create Deployment Engines

If you need to run tasks within *Windows PE* environment (through *BootOnWinPE* built-in action), then you must create *Deployment Engine* objects in Tivoli Provisioning Manager for OS Deployment.

A machine with Windows Automated Installation Kit (Windows AIK) and Windows Assessment and Deployment Kit (Windows ADK) installed, both in English version, is needed with a running Tivoli Provisioning Manager for OS Deployment agent (`rbagent` binary) on it. With the agent connected to the server, start the Web UI wizard from **Advanced Features → Deployment engines → Create New**.

Eventually ensure your model matching rules let your Deployment Engines cover all your target models.

Create your custom task

To start a custom *RemboC* script within Windows PE context as default action on all targets, a *Custom Task* object must be defined and the *Extension Point* pluggable logic must be used.

First step is the creation of the custom task on the Tivoli Provisioning Manager for OS Deployment server.

The following sample task includes two custom actions:

- action #0 (*BootOnWinPE*) starts Windows PE Deployment Engine previously created (assuming it is matching target model)
- action #1 (*RunCustomScript*) within this execution environment, it executes a RemboC scripts containing a given business logic

The custom task descriptor file (*hostmenu.ini*) is an INI formatted text file containing the list of actions composing the custom tasks.

The example below shows the "*hostmenu*" custom task made of 2 actions and their parameters:

```
[CUSTOM]
name=hostmenu
description=Hostmenu application
numactions=2

[Action_0]
name=BootOnWinPE

[Parameters_Action_0]
usePhysical=true
arch=x86

[Action_1]
name=RunCustomScript

[Parameters_Action_1]
script=hostmenu.rbc
```

File global\remboc\hostmenu.ini

Given the custom task descriptor file, you can create the task with the following command:

./rbagent -d -v 4 -s 127.0.0.1:xxxxx rad-createscheme /opt/tivoli/tpmfos/files/global/remboc/hostmenu.ini

The RemboC script being executed in the second action (*RunCustomScript*) is in charge of showing a user interface made of buttons executing given actions. Below a sample code block template showing a basic UI waiting for user input and running callback functions accordingly:

```
str sel = '';
while (true) {
   str status = getCustomActivityIniValue("status","step");
   if (status=="") {
       sel = DrawMenu();
       MessageBox(NIL, !!'Menu selection', Strf(!!'Action selected: \'%s\'',sel),'o');
       setStatus("step",sel);
   } else if (status=='off') {
      setStatus("step","");
      MessageBox(NIL, !!'Action message', Strf(!!'Powering off ...'),'o');
...
...
}
```

File `global\remboc\hostmenu.rbc`

The RemboC script base path is the *<DATADIR>\global\remboc* folder: all RemboC scripts are searched within this directory.

At this point a task named "*hostmenu*" is stored into the TPMfOSd server and it is ready to be scheduled on targets: the actions composing the activity will start Windows PE and execute the RemboC script.

Configure default tasks

Using a pluggable logic, it is possible to define and customize server and agent behaviors during some key phase of the deployment.

For kernel-free targets, it is possible to define the boot behavior with a RemboC file placed at the path *<DATADIR>\global\serverext*. This RemboC file must register a defined user function as extension point for the TARGET_BOOT pluggable phase. This is done through the function:

registerExtensionPoint(<extension_point_ID>,<description>,<function>)

For example, each kernel-free target runs the following function declared and registered in the file *<DATADIR>\global\serverext\plugin.rbc*

```
var extension_function(var params){
      var myini = NewIniFile();
      str defaultTask = "hostmenu";
      SetIniValue(myini, "TASK", "name", defaultTask);
      return myini;
}
//Register the extension function to manage the kernel-free targets during PXE-boot.
bool val = registerExtensionPoint('TARGET_BOOT','default_hostmenu',extension_function);
```

File `global\serverext\plugin.rbc`

This function modifies the kernel-free target boot forcing to run task “*hostmenu*” as default task.

The *<DATADIR>\global\serverext* folder is read every time the Tivoli Provisioning Manager for OS Deployment server is started: to take new changes remember to restart it.

Custom task execution

Once the environment has been prepared and tasks and engines are ready, targets can PXE-boot and executes the defined logic.

A common target behavior is made of the following sequence of steps:

- a network boot
- a download of a minimal context switcher binary that will load next requested execution engine (according to the pending task detected)
- a task scheduled or defined as default requesting the switcher chain-loading the selected engine on the target (Windows PE, Linux DE, Hard-Disk)
- the RemboC script running in the started execution environment

Run your custom task

Newly discovered targets performing a PXE-boot, will apply the kernel-free mode (assuming previously configured): so the Syslinux boot loader will be started.

1. **Target PXE-boot in kernel-free mode**

   A basic user interface will be shown while contacting the server for a pending task to be executed
2. **Pre-OS environment loading default task**

If a task has been scheduled (or default one configured previously) it will be executed by loading the requested environment: in this case the custom task requires the action *BootOnWinPE* which needs the environment Windows PE being started

3. **Windows PE downloaded from the network**

   A Tivoli Provisioning Manager for OS Deployment agent is automatically injected into the Windows PE engine in order to grab machine control at start-up:

4. **Windows PE starts**

   After the first action is completed, the task will continue with the next action: a custom RemboC script

5. **RemboC custom task loads**

   The RemboC script will show its own UI and handle user interaction according to its content.

6. **RemboC custom task runs**

Sysprep and Capture

Within a custom RemboC script, it may be needed to change the script execution environment by starting the machine from the hard-disk or booting a ramdisk (cached or from the network). As example having a custom user interface (UI) running under Windows PE, the user can select a "*Sysprep and Capture*" button thus expecting the target to boot on the disk, run `sysprep.exe` executable and then reboot back on Windows PE environment.

The section "Booting on the disk from RemboC scripts" describes how to change execution context within a remboC script by simulating a "sysprep and capture" scenario.

Booting on the disk from RemboC scripts

Starting from a RemboC script showing a custom UI, we may need to handle the user request to perform a *Sysprep and Capture*operation: this means the code behind the selected buttons must be able to start the machine from the disk, run the Tivoli Provisioning Manager for OS Deployment agent again to continue the custom task, start the `sysprep.exe` binary and reboot back in Windows PE showing again the custom UI.

The *Sysprep and Capture*operation comprises the following steps:

- When UI button is selected from Windows PE, Tivoli Provisioning Manager for OS Deployment agent must be registered to start from OS at next reboot and a boot on hard disk must be performed to continue the action from the OS
- When Tivoli Provisioning Manager for OS Deployment is running within the OS, the RemboC action must continue running `sysprep.exe` binary and rebooting back in Windows PE to complete the operation

The callback code running just after user interaction, may be similar to the following:

```
// Register Agent execution as RunOnce on OS image
registerRunOnceAgent();
// Boot on Hard-Disk built-in action
```

```
struct {} params;
params[(str)"diskno"] := (var)0;
params[(str)"partno"] := (var)1;
BootOnHD(params);
```

Sample code to invoke BootOnHD built-in action

This code must register Tivoli Provisioning Manager for OS Deployment agent to run at operating system startup in order to grab machine control and continue the task execution. The function `registerRunOnceAgent` is a sample function assumed to register Tivoli Provisioning Manager for OS Deployment agent at OS boot and as soon as the operating system is started, run the following command:

```
rbagent.exe -d -v 4 -s <IP>:<PWD> -e AGENT_WINSTALL rad-runtask
```

Then this code must run the built-in action *BootOnHD* as a primitive from the RemboC script itself: parameters are provided through the *struct* data type creating an hashed map of parameters. At requested machine boot, the operating system on the disk starts and the `RunOnce` script prepares the `rbagent` binary to run.

The registered `RunOnce` script connects Tivoli Provisioning Manager for OS Deployment agent to the server and continues the custom tasks execution: the pending RemboC action will be started again.

The system state must be internally handled by the RemboC script in order to switch execution environment (or context) and keep track of internal execution progresses. The RemboC script is always executed from the beginning of the source code, so tracking execution points must be internally implemented. To keep persistent data on files (activity files) the following methods can be used:

- *getCustomActivityIniValue(<SECTION>,<KEY>); // returns <VALUE>*
- *setCustomActivityIniStruct(<SECTION>,<KEY_VALUE_STRUCT>); // set values in <SECTION>*

For example, the following code can be used to read and write a persistent information related to the script execution state:

```
//
// INI sample
// [status]
// step=STEP1
//
struct {} values;
values[(str)"step"] = (var)"STEP1";
setCustomActivityIniStruct("status",values);
...
str status = getCustomActivityIniValue("status","step");
```

Next step performed by the script handling the *Sysprep and Capture* request, will be to run `sysprep.exe` binary once the RemboC script is running within the OS (no code sample is given to start an executable within a running OS, several ways are available).

Then the Tivoli Provisioning Manager for OS Deployment agent executes next code block in the RemboC Action invoking the *BootOnWinPE* built-in action as a primitive. For example:

```
// Boot on Windows PE built-in action
struct {} bootwpeparams;
BootOnWinPE(bootwpeparams);
```

Sample code to invoke BootOnWinPE built-in action

After triggering a *BootOnWinPE* command, the machine reboots on Windows PE to complete the capture and show the custom UI. System states have to be internally handled to in order to track the execution status of custom operations within the RemboC scripts.

Storing persistent task data

The following public methods store persistent information in the activity files: this files are kept during task execution and can be used within RemboC scripts to track information that must not be volatile:

getCustomActivityIniValue(<SECTION>,<KEY>); // returns <VALUE>

setCustomActivityIniStruct(<SECTION>,<KEY_VALUE_STRUCT>); // set values in <SECTION>

Each custom task is tracked with a dedicated folder containing task configuration and trace files. For task started from the TPMfOSd server, they are stored at path:

<DATADIR>\global\hostactivities\task<TASKID>

The *setCustomActivityIniStruct* can be used in this way

```
struct {} values;
values[(str)"KEY_1"] = (var)"VALUE_1";
setCustomActivityIniStruct("status",values);
```

The *getCustomActivityIniValue* can be used in this way to read that value

```
str value = getCustomActivityIniStruct("status","KEY_1");
```

Task INI file

```
[CUSTOM]
name=MyCustomTask
description=Testing everything
numactions=7
REBOOT_ON_FAILURE=false
password=XXXXXXXXXXXXXXXXXXXX
timeout=20

[Action_0]

name=BootOnWinPE

[Parameters_Action_0]

usePhysical=true

arch=x86

[Action_1]

name=RunCustomScript

[Parameters_Action_1]

script=myscript.rbc
```

```
[Action_2]

name=BootOnLDE

[Parameters_Action_2]

[Action_3]

name=BootOnHD

[Parameters_Action_3]

diskno=0

partno=0

[Action_4]

name=RunCustomScript

[Parameters_Action_4]

script=myscript.rbc

[Action_5]

name=BootOnWinPE

[Parameters_Action_5]

usePhysical=false

[Action_6]

name=RebootOnEnv

[Parameters_Action_6]

target_env=AGENT_WPE

usePhysical=true

arch=x86
```

Scripts with built-in actions used as primitive

The following are examples of a Rembo-C custom script.

The first example is a Rembo-C custom script that can be used when working in online mode to switch execution environments:

```
Printf("Entering my script\r\n");

if (getCustomActivityIniValue("status","step")=="") {
    setCustomActivityIniValue("status","step","1");
    Printf("Booting hd...\r\n");
    struct {} params;
    params[(str)"diskno"] := (var)0;
    params[(str)"partno"] := (var)1;
    BootOnHD(params);

} else if (getCustomActivityIniValue("status","step")=="1") {
    setCustomActivityIniValue("status","step","2");
    Printf("Booting wpe...\r\n");

    struct {} bootwpeparams;
    BootOnWinPE(bootwpeparams);

} else if (getCustomActivityIniValue("status","step")=="2") {
    setCustomActivityIniValue("status","step","3");
    Printf("Booting hd...\r\n");

    struct {} params;
    params[(str)"diskno"] := (var)0;
    params[(str)"partno"] := (var)1;
    BootOnHD(params);

} else if (getCustomActivityIniValue("status","step")=="3") {
    setCustomActivityIniValue("status","step","4");
    Printf("Booting Linux DE...\r\n");

    struct {} params;
    BootOnLDE(params);

} else if (getCustomActivityIniValue("status","step")=="4") {
    //Reboot on WPE
    struct {} parameters;
    parameters["target_env"] = "AGENT_WPE";
    parameters["usePhysical"] = "true";
    parameters["arch"] = "x86";
    RebootOnEnv(parameters);

} else if (getCustomActivityIniValue("status","step")=="5") {
    //Reboot on HD
    struct {} parameters;
    parameters["partno"] = "0";
    parameters["diskno"] = "0";
    RebootOnEnv(parameters);

} else if (getCustomActivityIniValue("status","step")=="6") {
    //Reboot on LDE
    struct {} parameters;
    parameters["target_env"] = "AGENT_LINUXDE";
    RebootOnEnv(parameters);
}

Printf("Exiting\r\n");
```

The second example is a Rembo-C custom script that can be used when working in online mode to

create the cache and copy the deployment engines on a target.

```
Printf("Entering test");

Printf("writing cache");
struct {} params;
params[(str)"target"] = (var)"disk://0:1/";
params[(str)"loader"] = (var)"RB";
WriteCache(params);
Printf("copied cache");

Printf("copying all the engines locally");
```

```
struct {} wpeparams;
wpeparams[(str)"target"] = (var)"enginewpe";
CacheWPE(wpeparams);
Printf("copied WPE");

Printf("Exiting from test");
```

The third example is a Rembo-C custom script that can be used when working in offline mode to switch execution environments. Cache and engines used were prepared previously with an online task.

```
Printf("Entering my script");

if (getCustomActivityIniValue("status","step")=="") {
    setCustomActivityIniValue("status","step","1");
    Printf("Booting wpe...");

    struct {} bootwpeparams;

    bootwpeparams[(str)"enginePath"] := (var)"enginewpe";

    BootOnWinPE(bootwpeparams);

} else if (getCustomActivityIniValue("status","step")=="1") {
    setCustomActivityIniValue("status","step","2");
    Printf("Booting hd...");

    struct {} params;
    params[(str)"diskno"] := (var)0;
    params[(str)"partno"] := (var)1;
    BootOnHD(params);

} else if (getCustomActivityIniValue("status","step")=="2") {
    setCustomActivityIniValue("status","step","3");

    //Reboot on WPE
    struct {} parameters;
    parameters["target_env"] = "AGENT_WPE";
    parameters["usePhysical"] = "true";
    parameters["arch"] = "x86";
    parameters["enginePath"] = "enginewpe";
    RebootOnEnv(parameters);

}

Printf("Exiting");
```

Chapter 13. Migrating users

When an operating system needs to be upgraded to a newer version or when hardware needs to be changed to newer material, users like to keep their settings and files and to have them available on their renewed computer. This is the purpose of *user migration*.

Basic operations for user migration are:

1. Performing an inventory of computer settings;
2. Capturing user settings to be migrated;
3. Reinstalling (or installing on a bare-metal computer) the operating system;
4. Restoring the computer and user settings on the target.

Tivoli Provisioning Manager for Images facilitates steps 1, 3, and 4 of this migration process.

Capturing user settings

Data to be captured on a computer for user migration includes, among others, user settings (user name, time zone, keyboard information), user files, and the list of installed software. The data needs to be retrieved and then stored outside of the source computer, typically on a network, in prevision of a future restoration.

Some of the needed data (including user accounts, users' files, desktop settings) can be captured with tools broadly available..

Other data (such as the list of installed software) can be captured through a command-line. The information thus obtained is then stored in the OS deployment server. The following command must be run on the source target.

```
rbagent -s serverIPaddress:NetPassword rad-hostinventory updatebom
```

where `serverIPaddress` is the IP address of the OS deployment server, `NetPassword` is the superuser password allowing the web interface extension to be connected to the OS deployment server. The password can naturally be provided in the same encrypted format as found in the configuration file `rembo.conf`. `rad-hostinventory` performs the inventory itself, while `uploadbom` modifies the target record on the OS deployment server.

Restoring user settings

Tivoli Provisioning Manager for Images enables you to restore previously captured user settings during a deployment task, with the help of software modules. The migration tool and the stored settings are included in software modules which are bound to a deployment scheme. The necessary software modules can be of several types, including registry changes and files copies. A careful ordering of the software modules is necessary as they are interdependent: the migration tool must obviously be installed and the setting files copied before the restoration operation is performed on the target.

Chapter 14. Glossary

A

administrative group
A group of related computers. An administrator can create administrative groups to organize target systems into meaningful categories, and to facilitate deployment of software to multiple targets.

B

bare metal computer
A computer on which there is nothing reliable but the hardware. It can be coming straight from factory without any data on its hard disk (out of the box) or it can contain a possibly damaged operating system.

Basic Input/Output System (BIOS)
The code that controls basic hardware operations, such as interactions with diskette drives, hard disk drives, and the keyboard.

BIOS See Basic Input/Output System.

blacklist
In Tivoli Provisioning Manager for Images, a list of PCI devices or of computer models which are known to raise issues, accompanied by hardware settings which must be used to work around the issues.

C

child An OS deployment server that is a subordinate of another OS deployment server in a replication tree structure. Only the top-level parent OS deployment server is not a child. See also parent.

clone To prepare a reference computer and create a system profile ready for deployment.

D

database server
The computer on which the database application and database are installed.

Deployment
A process which installs an operating system, and possibly other applications and files, on a target computer. During a deployment, data previously stored on the hard drives of the target is deleted.

Deployment scheme
A specific type of task template. A deployment scheme contains parameters for customizing a deployment on a target, and the target display screen layout. See also task template.

DHCP See Dynamic Host Configuration Protocol.

Dynamic Host Configuration Protocol (DHCP)
A communications protocol that is used to centrally manage configuration information. For example, DHCP automatically assigns IP addresses to computers in a network.

F

free-text condition
In Tivoli Provisioning Manager for Images, a condition written in Rembo-C; syntax, using variables and Java-like logical operators, and which evaluates to true or false.

H

hardware configuration
A set of parameters used to configure hardware before an operating system installation. It includes RAID settings, BIOS update information, BIOS settings, and custom hardware configuration parameters.

M

MCAST
A proprietary transfer protocol of Tivoli Provisioning Manager for Images computers using multicast. Contrast with unicast and PCAST.

MTFTP
See Multicast Trivial File Transfer Protocol.

multicast
Bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to many computers.

Multicast Trivial File Transfer Protocol (MTFTP)
Multicast TFTP.

N

network boot
The process of starting up a computer directly over the network rather than on a disk.

O

OS configuration
The operating system parameters of a system profile .

OS deployment server
The computer on which the Tivoli Provisioning Manager for Images application and files are installed.

P

parent An OS deployment server in a replication tree structure that has at least one dependent OS deployment server. See also child.

PCAST
A proprietary transfer protocol of Tivoli Provisioning Manager for Images that delivers non-identical sets of files to several target computers using multicast. Contrast with MCAST and unicast.

PCI See Peripheral Component Interconnect.

Peripheral Component Interconnect
A local bus that provides a high-speed data path between the processor and attached devices.

Preboot Execution Environment (PXE)
PXE is an industry standard target/server interface that allows networked

computers that are not yet loaded with an operating system to be configured and booted remotely. PXE is based on Dynamic Host Configuration Protocol (DHCP). Using the PXE protocol, targets can request configuration parameter values and startable images from the server. The PXE process consists of the system initiating the protocol by broadcasting a DHCPREQUEST containing an extension that identifies the request as coming from a target that uses PXE. The server sends the target a list of OS deployment servers that contain the operating systems available. The target then selects and discovers an OS deployment server and receives the name of the executable file on the chosen OS deployment server. The target downloads the file using Trivial File Transfer Protocol (TFTP) and runs it, which loads the operating system.

PXE See Preboot Execution Environment.

R

RAD file
A file containing deployment objects such as task templates, system profiles, and software modules used to archive data or to transfer data between two OS deployment servers. A RAD file has a `.rad` extension.

RAID See Redundant Array of Independent Disks.

redeployment
The process of synchronizing a hard-disk content to its reference image stored on a hidden and protected redeployment partition.

redeployment preload
The process of creating a reference image of a computer at the end of a deployment, and saving this reference image into a protected redeployment partition (invisible to the user and to the operating system itself).

Redundant Array of Independent Disks (RAID)
RAID is a way of storing the same data in different places (thus, redundantly) on multiple hard disks. By placing data on multiple disks, I/O operations can overlap in a balanced way, improving performance. Multiple disks increase the mean time between failure (MTBF) and storing data redundantly increases fault-tolerance.

Rembo-C;
A programming language, descendant of the C language combined with traces of JavaScript and Java.

replicated server
An OS deployment server which shares data with one or several other OS deployment servers. The servers are hierarchically structured with a parent and child servers. A child child can act as parent to replicated servers further down in the hierarchy.

replication
The process of copying files from a parent server to a child server. A selection can be performed on the kind of information that must be replicated. Files that have been modified are copied over.

S

shared repository
In Tivoli Provisioning Manager for Images, a repository of server objects

where each file is stored only once, even if it belongs to several objects. The shared repository reduces the storage space necessary to hold all server objects.

software module
A group of files, and potentially command lines, packaged together under one name. A software module can be installed on a target during a deployment.

software snapshot
A differential image of software installed on top of a running operating system. Software snapshot creation is deprecated. Any previously created software snapshots can be deployed for compatibility with earlier versions.

system profile
The partition layout and list of files for deployment of an operating system, either by unattended setup or by cloning. A system profile can have several configurations.

system snapshot
For Windows only. The partition layout and list of files for deployment of an operating system, created by cloning without using Sysrep. A system snapshot cannot be parametrized and can only be restored, not deployed.

T

target A computer that is known to an OS deployment server.

target list
A comma-separated-value list of targets used for adding large numbers of targets to the OS deployment server without having to start the targets up individually on the network.

task A set of actions designed to achieve a particular result. A task is performed on a set of targets on a specific schedule.

task template
A group of elements which can be customized on a target computer. These elements are mostly screen layouts which condition the appearance of the target computer screen during the different phases of its control by Tivoli Provisioning Manager for Images. See also Deployment scheme.

TCP tunnel
A way to provide TCP connectivity to target computers.

TFTP See Trivial File Transfer Protocol.

Trivial File Transfer Protocol (TFTP)
In Internet communications, a set of conventions that transfers files between targets using minimal protocol.

U

unattended setup
Operating system installation on a target, using original installation files and parameters contained in a script defined on the OS deployment server. Contrast with clone.

unicast
Transmission of data to a single destination. In Tivoli Provisioning Manager for Images, a transfer protocol that delivers a stream of files to a single target. Based on TCP, this protocol is faster when there are only a

few target computers on the receiving end of the transfer. This protocol can also be used in networks where multicast traffic is not properly handled. Contrast with MCAST and PCAST.

universal image
A cloned system profile that has been prepared with all drivers for disk types and hardware abstraction layer variants encountered in the pool of targets to be deployed.

W

Wake on LAN
A technology that enables a user to remotely turn on systems for off-hours maintenance. A result of the Intel-IBM Advanced Manageability Alliance and part of the Wired for Management Baseline Specification, users of this technology can remotely turn on a server and control it across the network, thus saving time on automated software installations, upgrades, disk backups, and virus scans.

Web interface
A user interface for one or more administrative tasks.

Web interface extension
An agent that allows the web interface to have access to the content of the target on which it is running. For example, to browse disks and read and write files.

Z

zone An IP range or domain that is used to logically group computers into regions. You can define one or more zones for each region.

Chapter 15. Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM‘s application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Notice for Windows Automated Installation Kit (AIK)

Windows Automated Installation Kit (AIK) for Windows 7 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

The Windows AIK is licensed to you by the code's owner and not by IBM it is your responsibility to determine whether the license terms offered by the code's owner are acceptable to you.

YOUR USE OF THE WAIK AND ANY URL'S OR MATERIALS ON THIRD PARTY WEBSITES ("THIRD PARTY MATERIALS") IS "AS IS", WITHOUT WARRANTY FROM IBM OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. TO THE EXTENT PERMITTED BY LAW, IBM DISCLAIMS ALL LIABILITY FOR ANY CLAIMS ARISING OUT OF USE OF THE THIRD PARTY MATERIALS.

Notice for Windows Assessment and Deployment Kit (ADK) for Windows 8

Windows Assessment and Deployment Kit (ADK) for Windows 8 in English is distributed by Microsoft and is available on the Microsoft website from the following link at the time of publication: http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en.

The Windows AIK is licensed to you by the code's owner and not by IBM it is your responsibility to determine whether the license terms offered by the code's owner are acceptable to you.

YOUR USE OF THE WAIK AND ANY URL'S OR MATERIALS ON THIRD PARTY WEBSITES ("THIRD PARTY MATERIALS") IS "AS IS", WITHOUT WARRANTY FROM IBM OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. TO THE EXTENT PERMITTED BY LAW, IBM DISCLAIMS ALL LIABILITY FOR ANY CLAIMS ARISING OUT OF USE OF THE THIRD PARTY MATERIALS.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml

Adobe is either a registered trademark or trademark of Adobe Systems Incorporated in the United States, other countries, or both.

Intel and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows , and Windows NT are trademarks or registered trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States, other countries, or both.

Other company, product and service names may be trademarks or service marks of others.

Copyrights

IBM web site pages may contain other proprietary notices and copyright information which should be observed.

Portions of third-party software included in this IBM product is used with permission and is covered under the following copyright attribution statements:

- Copyright (c) 1998-2005, The OpenSSL Project. All rights reserved.
- Copyright (c) 1995-2005 Jean-loup Gailly and Mark Adler, the ZLIB data compression library.
- Copyright 1994-2006, The FreeBSD Project. All rights reserved.

The MD5 Message-Digest Algorithm was developed by Ron Rivest. The public domain C language implementation used in this program was written by Colin Plumb in 1993. Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, without any conditions or restrictions. This software is provided "as is" without express or implied warranty.

Portions include cryptographic software written by Eric Young (<(eay@cryptosoft.com)>). This product may include software written by Tim Hudson (<(tjh@cryptosoft.com)>).

Printed in USA