Table of Contents

Introduction to operating system deployment in System Center Configuration Manager

11/23/2016 • 8 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can use Configuration Manager to deploy operating systems in a number of different ways. Use the information in this section to understand how to deploy operating systems and automate tasks.

The operating system deployment process

Configuration Manager provides several methods that you can use to deploy an operating system. There are several actions that you must take regardless of the deployment method that you use:

- Identify Windows device drivers that are required to start the boot image or install the operating system image that you have to deploy.
- Identify the boot image that you want to use to start the destination computer.
- Use a task sequence to capture an image of the operating system that you will deploy. Alternatively, you can use a default operating system image.
- Distribute the boot image, operating system image, and any related content to a distribution point.
- Create a task sequence with the steps to deploy the boot image and the operating system image.
- Deploy the task sequence to a collection of computers.
- Monitor the deployment.

Operating system deployment scenarios

There are many operating system deployment scenarios in Configuration Manager that you can choose from depending on your environment and the purpose for the operating system installation. For example, you can partition and format an existing computer with a new version of Windows or upgrade Windows to the latest version. To help you determine the deployment method that meets your needs, review Scenarios to deploy enterprise operating systems. You can choose from the following operating system deployment scenarios:

- Upgrade Windows to the latest version
- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)
- Replace an existing computer and transfer settings

Methods to deploy operating systems

There are several methods that you can use to deploy operating systems to Configuration Manager client computers.

- **PXE initiated deployments**: PXE-initiated deployments let client computers request a deployment over the network. In this method of deployment, the operating system image and a Windows PE boot image are sent to a distribution point that is configured to accept PXE boot requests. For more information, see Use PXE to

deploy Windows over the network with System Center Configuration Manager.

- **Make operating systems available in Software Center**: You can deploy an operating system and make it available in the Software Center. Configuration Manager clients can initiate the operating system installation from Software Center. For more information, see Replace an existing computer and transfer settings.
- **Multicast deployments**: Multicast deployments conserve network bandwidth by concurrently sending data to multiple clients instead of sending a copy of the data to each client over a separate connection. In this method of deployment, the operating system image is sent to a distribution point. This in turn deploys the image when client computers request the deployment. For more information, see Use multicast to deploy Windows over the network.
- **Bootable media deployments**: Bootable media deployments let you deploy the operating system when the destination computer starts. When the destination computer starts, it retrieves the task sequence, the operating system image, and any other required content from the network. Because that content is not included on the media, you can update the content without having to re-create the media. For more information, see Create bootable media.
- **Stand-alone media deployments**: Stand-alone media deployments let you deploy operating systems in the following conditions:
  - In environments where it is not practical to copy an operating system image or other large packages over the network.
  - In environments without network connectivity or low bandwidth network connectivity.

  For more information, see Create stand-alone media.
- **Pre-staged media deployments**: Pre-staged media deployments let you deploy an operating system to a computer that is not fully provisioned. The pre-staged media is a Windows Imaging Format (WIM) file that can be installed on a bare-metal computer by the manufacturer or at an enterprise staging center that is not connected to the Configuration Manager environment.

  Later in the Configuration Manager environment, the computer starts by using the boot image provided by the media, and then connects to the site management point for available task sequences that complete the download process. This method of deployment can reduce network traffic because the boot image and operating system image are already on the destination computer. You can specify applications, packages, and driver packages to include in the pre-staged media. For more information, see Create prestaged media.

Boot images

A boot image in Configuration Manager is a Windows PE (WinPE) image that is used during an operating system deployment. Boot images are used to start a computer in WinPE, which is a minimal operating system with limited components and services that prepare the destination computer for Windows installation. Configuration Manager provides two boot images: One to support x86 platforms and one to support x64 platforms. These are considered default boot images. Boot images that you create and add to Configuration Manager are considered custom images. Default boot images can be automatically replaced when you update Configuration Manager. For more information about boot images, see Manage boot images.

Operating system images

Operating system images in Configuration Manager are stored in the Windows Imaging (WIM) file format and represent a compressed collection of reference files and folders that are required to successfully install and configure an operating system on a computer. For all operating system deployment scenarios, you must select an operating system image. You can use the default operating system image or build the operating system image from a reference computer that you configure. For more information, see Manage operating system images.

Operating system upgrade packages

Operating system upgrade packages are used to upgrade an operating system and are setup-initiated operating system deployments. You import operating system upgrade packages to Configuration Manager from a DVD or mounted ISO file. For more information, see Manage operating system upgrade packages.

Media to deploy operating systems

You can create several kinds of media that can be used to deploy operating systems. This includes capture media that is used to capture operating system images and stand-alone, pre-staged, and bootable media that is used to deploy an operating system. By using media, you can deploy operating systems on computers that do not have a network connection or that have a low bandwidth connection to your Configuration Manager site. For more information about how to use media, see Create task sequence media.

Device drivers

You can install device drivers on destination computers without including them in the operating system image that is being deployed. Configuration Manager provides a driver catalog that contains references to all the device drivers that you import into Configuration Manager. The driver catalog is located in the **Software Library** workspace and consists of two nodes: **Drivers** and **Driver Packages**. The **Drivers** node lists all the drivers that you have imported into the driver catalog. You can use this node to discover the details about each imported driver, to change what driver package or boot image a driver belongs to, to enable or disable a driver, and more. For more information, see Manage drivers.

Save and restore user state

When you deploy operating systems, you can save the user state from the destination computer, deploy the operating system, and then restore the user state after the operating systems is deployed. This process is typically used when you install the operating system on a Configuration Manager client computer.

The user state information is captured and restored by using task sequences. When the user state information is captured, the information can be stored in one of the following ways:

- You can store the user state data remotely by configuring a state migration point. The Capture task sequence sends the data to the state migration point. Then, after the operating system is deployed, the Restore task sequence retrieves the data and restores the user state on the destination computer.
- You can store the user state data locally to a specific location. In this scenario, the Capture task sequence copies the user data to a specific location on the destination computer. Then, after the operating system is deployed, the Restore task sequence retrieves the user data from that location.
- You can specify hard links that can be used to restore the user data to its original location. In this scenario, the user state data remains on the drive when the old operating system is removed. Then, after the operating system is deployed, the Restore task sequence uses the hard links to restore the user state data to its original location.

For more information Manage user state.

Deploy to unknown computers

You can deploy an operating system to computers that are not managed by Configuration Manager. There is no record of these computers in the Configuration Manager database. These computers are referred to as unknown computers. Unknown computers include the following:

- A computer where the Configuration Manager client is not installed

- A computer that is not imported into Configuration Manager
- A computer that is not discovered by Configuration Manager

For more information, see Prepare for unknown computer deployments.

Associate users with a computer

When you deploy an operating system, you can associate users with the destination computer to support user device affinity actions. When you associate a user with the destination computer, the administrative user can later perform actions on whichever computer is associated with that user, such as deploying an application to the computer of a specific user. However, when you deploy an operating system, you cannot deploy the operating system to the computer of a specific user. For more information, see Associate users with a destination computer.

Use task sequences to automate steps

You can create task sequences to perform a variety of tasks within your Configuration Manager environment. The actions of the task sequence are defined in the individual steps of the sequence. When the task sequence is run, the actions of each step are performed at the command-line level without requiring user intervention. You can use task sequences for the following:

- Create a task sequence to install an operating system
- Create a task sequence for non-operating system deployments
- Create a task sequence to capture an operating system
- Create a task sequence to capture and restore user state
- Create a custom task sequence

Task sequence steps in System Center Configuration Manager

3/26/2017 • 82 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

The following task sequence steps can be added to a Configuration Manager task sequence. For information about editing a task sequence, see Edit a task sequence.

Apply Data Image Task Sequence Step

Use the **Apply Data Image** task sequence step to copy the data image to the specified destination partition.

This step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Task sequence action variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name

A short user-defined name that describes the action taken in this step.

Description

More detailed information about the action taken in this step.

Image Package

Specify the **Image Package** that will be used by this task sequence step by clicking **Browse**. Select the package you want to install in the **Select a Package** dialog box. The associated property information for each existing image package is displayed at the bottom of the **Select a Package** dialog box. Use the drop-down list to select the **Image** you want to install from the selected **Image Package**.

> **NOTE**
> This task sequence action treats the image as a data file and does not do any of the setup necessary to boot the image as an operating system.

Destination

Specifies an existing formatted partition and hard disk, specific logical drive letter, or the name of a task sequence variable that contains the logical drive letter.

- **Next available partition** - Use the next sequential partition that has not been previously targeted by an Apply Operating System or Apply Data Image action in this task sequence.
- **Specific disk and partition** - Select the **Disk** number (starting with 0) and the **Partition** number (starting with 1).

- **Specific logical drive letter** - Specify the **Drive Letter** assigned to the partition by Windows PE. Note that this drive letter can be different from the drive letter that the newly deployed operating system will assign.
- **Logical drive letter stored in a variable** - Specify the task sequence variable containing the drive letter assigned to the partition by Windows PE. This variable would typically be set in Advanced section of the **Partition Properties** dialog box for the **Format and Partition Disk** task sequence action.

Delete all content on the partition before applying the image
Specifies that all files on the target partition will be deleted before the image is installed. By not deleting the content of the partition, this step can be used to apply additional content to a previously targeted partition.

Apply Driver Package

Use the **Apply Driver Package** task sequence step to download all of the drivers in the driver package and install them on the Windows operating system.

The **Apply Driver Package** task sequence step makes all device drivers in a driver package available for use by Windows. This step can be added to a task sequence between the **Apply Operating System** and the **Setup Windows and ConfigMgr** steps to make the device drivers in the driver package available to Windows. Typically, the **Apply Driver Package** step is placed after the **Auto Apply Drivers** task sequence step. The **Apply Driver Package** task sequence step is also useful with stand-alone media deployment scenarios.

Ensure that similar device drivers are put into a driver package and distribute them to the appropriate distribution points. After they are distributed Configuration Manager client computers can install them. For example, you can put all the device drivers from a manufacturer into a driver package, and then distribute the package to distribution points where the associated computers can access them.

This step is useful for stand-alone media and for administrators who want to install a specific set of drivers, including drivers for devices that would not be detected in a Plug-n-Play scan (for example, network printers).

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Apply Driver Package Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Driver package
Specify the driver package that contains the needed device drivers by clicking **Browse** and launching the **Select a Package** dialog box. Specify an existing package to be made available. The associated package properties are displayed at the bottom of the dialog box.

Select the mass storage driver within the package that needs to be installed before setup on pre-Windows Vista operating systems

Specify any mass storage device drivers that are needed for pre- Windows Vista operating system installations.

Driver
Select the mass storage device driver file to be installed before setup on pre-Windows Vista operating system deployments. The drop-down list is populated from the specified package.

Model
Specify the boot-critical device that is needed for pre-Windows Vista operating system deployments.

Do unattended installation of unsigned drivers on version of Windows where this is allowed
Select this option to allow Windows to install drivers that are unsigned on the reference computer.

Apply Network Settings Step

Use the **Apply Network Settings** task sequence step to specify the network or workgroup configuration information for the destination computer. The specified values are stored in the appropriate answer file format for use by Windows Setup when the **Setup Windows and ConfigMgr** task sequence step is run.

This task sequence step runs in either a standard operating system or Windows PE. For more information about the task sequence variables for this action, see Apply Network Settings Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Join a workgroup
Select this option to have the destination computer join the specified workgroup. Enter the name of the workgroup on the **Workgroup** line. This value can be overridden by the value that is captured by the **Capture Network Settings** task sequence step.

Join a domain
Select this option to have the destination computer join the specified domain. Specify or browse to the domain, such as *fabricam.com*. Specify or browse to a Lightweight Directory Access Protocol (LDAP) path for an organizational unit (i.e. LDAP//OU=computers, DC=Fabricam.com, C=com).

Account
Click **Set** to specify an account with the necessary permissions to join the computer to the domain. In the **Windows User Account** dialog box you can enter the user name using the following format: **Domain\User** .

Adapter settings
Specify network configurations for each network adapter in the computer. Click **New** to open the **Network Settings** dialog box, and then specify the network settings. If network settings were captured in a previous **Capture Network Settings** task sequence step, the previous settings are applied to the network adapter and the settings specified in this in this step are not applied. If network settings were not previously captured, the settings specified in the **Apply Network Settings** step are applied to network adapters in Windows device enumeration order.

Apply Operating System Image

Use the **Apply Operating System Image** task sequence step to install an operating system on the destination computer. This task sequence step performs a set of actions depending on whether it is using an operating system image or an operating system installation package to install the operating system.

The **Apply Operating System Image** step performs the following actions when an operating system image is used.

1. Deletes all content on the targeted volume except for those files under the folder specified by the _SMSTSUserStatePath task sequence variable.
2. Extracts the contents of the specified .wim file to the specified destination partition.
3. Prepares the answer file:
    a. Creates a new default Windows Setup answer file (sysprep.inf or unattend.xml) for the operating system that is being deployed.
    b. Merges any values from the user-supplied answer file.
4. Copies Windows boot loaders into the active partition.
5. Sets up the boot.ini or the Boot Configuration Database (BCD) to reference the newly installed operating system.

The **Apply Operating System Image** step performs the following actions when an operating system installation package is used.

1. Deletes all content on the targeted volume except for those files under the folder specified by the _SMSTSUserStatePath task sequence variable.
2. Prepares the answer file:
    a. Creates a fresh answer file with standard values created by Configuration Manager.
    b. Merges any values from the user-supplied answer file.

> **NOTE**
> Actual installation of Windows is started by the **Setup Windows and ConfigMgr** task sequence step. After the **Apply Operating System** task sequence action has run, the OSDTargetSystemDrive task sequence variable is set to the drive letter of the partition containing the operating system files.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Apply Operating System Image Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- **Access content directly from the distribution point**:

  Use this option to specify whether you want the task sequence to access the operating system image directly from the distribution point. For example, you can use this option when you deploy operating

systems to embedded devices that have limited storage capacity. When this option is selected, you must also configure the package share settings on the **Data Access** tab of the package properties.

> **NOTE**
>
> This setting overrides the deployment option that is configured on the **Distribution Points** page in the **Deploy Software Wizard** only for the operating system image specified in this task sequence step, and not all content for the entire task sequence.

- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Apply operating system from a captured image
Installs an operating system image that has previously been captured. Click **Browse** to open the **Select a package** dialog box, and then select the existing image package you want to install. If multiple images are associated with the specified **Image package**, use the drop-down list to specify the associated image that will be used for this deployment. You can view basic information about each existing image by clicking on the image.

Apply operating system image from an original installation source
Installs an operating system using an original installation source. Click **Browse** to open the **Select and Operating System Install Package** dialog box, and then select the existing operating system installation package you want to use. You can view basic information about each existing image source by clicking on the image source. The associated image source properties are displayed in the results pane at the bottom of the dialog box. If there are multiple editions associated with the specified package, use the drop-down list to specify the associated **Edition** that is used.

Use an unattended or sysprep answer file for a custom installation
Use this option to provide a Windows setup answer file (**unattend.xml**, **unattend.txt**, or **sysprep.inf**) depending on the operating system version and installation method. The file you specify can include any of the standard configuration options supported by Windows answer files. For example, you can use it to specify the default Internet Explorer home page. You must specify the package that contains the answer file and the associated path to the file in the package.

> **NOTE**
>
> The Windows setup answer file that you supply can contain embedded task sequence variables of the form %*varname*%, where varname is the name of the variable. The %*varname*% string will be substituted for the actual variable values in the **Setup Windows and ConfigMgr** task sequence action. Note however, that such embedded task sequence variables cannot be used in numeric-only fields in an unattend.xml answer file.

If you do not supply a Windows setup answer file, this task sequence action will automatically generate an answer file.

Destination
Specifies an existing formatted partition and hard disk, specific logical drive letter, or the name of a task sequence variable that contains the logical drive letter.

- **Next available partition** - Use the next sequential partition that has not been previously targeted by an

Apply Operating System or Apply Data Image action in this task sequence.

- **Specific disk and partition** - Select the **Disk** number (starting with 0) and the **Partition** number (starting with 1).
- **Specific logical drive letter** - Specify the **Drive Letter** assigned to the partition by Windows PE. Note that this drive letter can be different from the drive letter that the newly deployed operating system will assign.
- **Logical drive letter stored in a variable** - Specify the task sequence variable containing the drive letter assigned to the partition by Windows PE. This variable would typically be set in Advanced section of the **Partition Properties** dialog box for the **Format and Partition Disk** task sequence action.

Apply Windows Settings

Use the **Apply Windows Settings** task sequence step to configure the Windows settings for the destination computer. The specified values are stored in the appropriate answer file format for use by Windows Setup when the **Setup Windows and ConfigMgr** task sequence step is run.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Apply Windows Settings Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step

Description
More detailed information about the action taken in this step.

User name
Specify the registered user name that is associated with the destination computer. This value can be overridden by the value that is captured by the **Capture Windows Settings** task sequence action.

Organization name
Specify the registered organization name that is associated with the destination computer. This value can be overridden by the value that is captured by the **Capture Windows Settings** task sequence action.

Product key
Specify the product key that is used for the Windows installation on the destination computer.

Server licensing
Specify the server licensing mode. You can select **Per server** or **Per user** as the licensing mode. If you select per Server as the licensing mode you will also need to specify the maximum number of connections that will be permitted per your license agreement. Select **Do not specify** if the destination computer is not a server or you do not want to specify the licensing mode.

Maximum connections
Specify the maximum number of connections that are available for this computer as stated in your license agreement.

Randomly generate the local administrator password and disable the account on all supported platforms (recommended)
Select this option to randomly generate a local administrator password. This creates a local administrator password and causes the account to be disabled on supported platforms.

Enable the account and specify the local administrator password
Select this option to enable the local administrator account and create the local administrator password. Enter the password on the **Password** line and confirm the password on the **Confirm password** line.

Time Zone
Specify the time zone to configure on the destination computer. This value can be overridden by the value that is captured by the **Capture Windows Settings** task sequence step.

Auto Apply Drivers

Use the **Auto Apply Drivers** task sequence step to match and install drivers as part of the operating system deployment.

The **Auto Apply Drivers** task sequence step performs the following actions:

1. Scans the hardware and finds the Plug-n-Play IDs for all devices present on the system.
2. Sends the list of devices and their Plug-n-Play IDs to the management point. The management point returns a list of compatible drivers from the driver catalog for each device. The management point considers all drivers regardless of what driver package they might be in. Only those drivers tagged with the specified driver category and those drivers that are not marked as disabled are considered.
3. For each device, the client picks the best driver that is appropriate for the operating system on which it is being deployed and that is on an accessible distribution point.
4. The selected driver or drivers are downloaded from a distribution point and staged on the target operating system.
    a. For image-based installations, the drivers are placed into the operating system driver store.
    b. For setup-based installations, Windows Setup is configured with where to find the drivers.
5. When the **Setup Windows and ConfigMgr** task sequence action runs and Windows initially boots, it will find the drivers staged by this action.

> **IMPORTANT**
> The **Auto Apply Drivers** task sequence step cannot be used with stand-alone media because Windows Setup will have no connection to the Configuration Manager site.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Auto Apply Drivers Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.

- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Install only the best matched compatible drivers
Specifies that the task sequence step installs only the best matched driver for each hardware device detected.

Install all compatible drivers
Specifies that the task sequence step installs all compatible drivers for each hardware device detected and allows Windows setup to choose the best driver. This option takes more network bandwidth and disk space because it downloads more drivers, but it can result in a better driver being selected.

Consider drivers from all categories
Specifies that the task sequence action searches all available driver categories for the appropriate device drivers.

Limit driver matching to only consider drivers in selected categories
Specifies that the task sequence action searches for device drivers in specified driver categories for the appropriate device drivers.

Do unattended installation of unsigned drivers on versions of Windows where this is allowed
Allows this task sequence action to install unsigned Windows device drivers.

> **IMPORTANT**
> This option does not apply to operating systems where driver signing policy cannot be configured.

Capture Network Settings

Use the **Capture Network Settings** task sequence step to capture Microsoft network settings from the computer running the task sequence. The settings are saved in task sequence variables that will override the default settings you configure on the **Apply Network Settings** task sequence step.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For more information about the task sequence variables for this action, see Capture Network Settings Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the action taken in this step.

Description
Provides more detailed information about the action taken in this step.

Migrate domain and workgroup membership

Captures the domain and workgroup membership information of the destination computer.

Migrate network adapter configuration
Captures the network adapter configuration of the destination computer. The captured information includes the global network settings, the number of adapters, and the network settings associated with each adapter. These settings include settings associated with DNS, WINS, IP, and port filters.

Capture Operating System Image

Use the **Capture Operating System Image** task sequence step to capture one or more images from a reference computer and store them in a WIM file on the specified network share. The Add Operating System Image Package Wizard can then be used to import this .WIM file into Configuration Manager so that it can be used for image-based operating system deployments.

Each volume (drive) on the reference computer is captured as a separate image within the .wim file. If the referenced computer has multiple volumes, the resulting WIM file will contain a separate image for each volume. Only volumes that are formatted as NTFS or FAT32 are captured. Volumes with other formats and USB volumes are skipped.

The installed operating system on the reference computer must be a version of Windows that is supported by Configuration Manager and must have been prepared by using the SysPrep tool. The installed operating system volume and the boot volume must be the same volume.

You must also enter a Windows account that has write permissions to the network share that you selected.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Capture Operating System Image Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Target
File system pathname to the location that Configuration Manager uses when storing the captured operating system image.

Description
An optional user-defined description of the captured operating system image that is stored in the .WIM file.

Version
An optional user-defined version number to assign to the captured operating system image. This value can be any combination of letters and numbers and is stored in the .WIM file.

Created by
The optional name of the user that created the operating system image and is stored in the WIM file.

Capture operating system image account
You must enter the Windows account that has permissions to the network share you specified. Click **Set** to specify the name of that Windows account.

Capture User State

Use the **Capture User State** task sequence step to use the User State Migration Tool (USMT) to capture user state and settings from the computer running the task sequence. This task sequence step is used in conjunction with the **Restore User State** task sequence step. With USMT 3.0.1 and later, this option always encrypts the USMT state store by using an encryption key generated and managed by Configuration Manager.

For more information about managing the user state when deploying operating systems, see Manage user state.

You can also use the **Capture User State** task sequence step with the **Request State Store** and **Release State Store** task sequence steps if you want to save the state settings to or restore settings from a state migration point in the Configuration Manager site.

The **Capture User State** task sequence step provides control over a limited subset of the most commonly used USMT options. Additional command-line options can be specified using the OSDMigrateAdditionalCaptureOptions task sequence variable.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Capture User State Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

User state migration tool package
Enter the Configuration Manager package that contains the version of USMT for this task sequence step to use when capturing the user state and settings. This package does not require a program. When the task sequence step is run, the task sequence will use the version of USMT in the package you specify. Specify a package containing the 32-bit or x64 version of USMT depending upon the architecture of the operating system from which you are capturing the state.

Capture all user profiles with standard options
Select this option to migrate all user profile information. This option is selected by default.

If you select this option, but do not select the option to Restore local computer user profiles in the Restore User State task sequence step, the task sequence will fail because Configuration Manager cannot migrate the new accounts without assigning them passwords. Also, if you use the **New Task Sequence** wizard and create a task sequence to **Install an existing image package**, the resulting task sequence defaults to Capture all user profiles with standard options, but does not select the option to Restore local computer user profiles (i.e. non-domain accounts).

Select **Restore local computer user profiles** and provide a password for the account to be migrated. In a manually created task sequence, this setting is found under the Restore User State step. In a task sequence created by the **New Task Sequence** wizard, this setting is found under the step **Restore User Files and Settings** wizard page.

If you have no local user accounts, this does not apply.

Customize how user profiles are captured
Select this option to specify a custom profile file migration. Click **Files** to select the configuration files for USMT to use with this step. You must specify a custom .xml file that contains rules that define the user state files to migrate.

Click here to select configuration files:
Select this option to select the configuration files in the USMT package you want to use for capturing user profiles. Click the **Files** button to launch the **Configuration Files** dialog box. To specify a configuration file, enter the name of the file on the **Filename** line and click the **Add** button.

Enable verbose logging
Enable this option to generate more detailed log file information. When capturing state, the log Scanstate.log is generated and stored in the task sequence Log folder in the \windows\system32\ccm\logs folder by default.

Skip files using encrypted file system
Enable this option if you want to skip capturing files that are encrypted with the Encrypted File System (EFS), including profile files. Depending on the operating system and the USMT version, encrypted files might not be readable after you restore. For more information, see the USMT documentation.

Copy by using file system access
Enable this option to specify any of the following settings:

- **Continue if some files cannot be captured**: Enable this setting to continue the migration process even if some files cannot be captured. If you disable this option, if a file cannot be captured then the task sequence step will fail. This option is enabled by default.
- **Capture locally by using links instead of by copying files**: Enable this setting to use NTFS hard-links to capture files.

  For more information about migrating data using hard-links, see Hard-Link Migration Store
- **Capture in off-line mode (Windows PE only)**: Enable this setting to capture the user state while in Windows PE instead of the full operating system.

Capture by using Volume Copy Shadow Services (VSS)
This option allows you to capture files even if they are locked for editing by another application.

Capture Windows Settings

Use the **Capture Windows Settings** task sequence step to capture the Windows settings from the computer running the task sequence. The settings are saved in task sequence variables that will override the default settings you configure on the **Apply Windows Settings** task sequence step.

This task sequence step runs in either Windows PE or a standard operating system. For more information about the task sequence variables for this action, see Capture Windows Settings Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.

- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Migrate computer name
Select this option to capture the NetBIOS computer name of the computer.

Migrate registered user and organization names
Select this option to capture the registered user and organization names from the computer.

Migrate time zone
Select this option to capture the time zone setting on the computer.

Check Readiness

Use the **Check Readiness** task sequence step to verify that the target computer meets the specified deployment prerequisite conditions.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step. For this step, do not select this setting or the step will only log the readiness checks and not stop the task sequence when a check fails.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Ensure minimum memory (MB)
Select this setting to verify the amount of memory, in megabytes, installed on the target computer meets or exceeds the amount specified. By default, this setting is selected.

Ensure minimum processor speed (MHz)
Select this setting to verify that the speed of the processor, in megahertz (MHz), installed in the target computer meets or exceeds the amount specified. By default, this setting is selected.

Ensure minimum free disk space (MB)
Select this setting to verify that the amount of free disk space, in megabytes, on the target computer meets or exceeds the amount specified.

Ensure current OS to be refreshed is
Select this setting to verify that the operating system installed on the target computer meets the requirement that you specify. By default, this setting is selected with a value of **CLIENT**.

Connect To Network Folder

Use the **Connect to Network Folder** task sequence action to create a connection to a shared network folder.

This task sequence step runs in a standard operating system or Windows PE. For more information about the task sequence variables for this action, see Connect to Network Folder Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Convert Disk to Dynamic

Use the **Convert Disk to Dynamic** task sequence step to convert a physical disk from a basic disk type to a dynamic disk type.

This step runs in either a standard operating system or Windows PE. For more information about the task sequence variables for this action, see Convert Disk to Dynamic Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Disk Number
The physical disk number of the disk that will be converted.

Disable BitLocker

Use the **Disable BitLocker** task sequence step to disable the BitLocker encryption on the current operating system drive, or on a specific drive. This action leaves the key protectors visible in clear text on the hard drive, but it does not decrypt the contents of the drive. Consequently this action is completed almost instantly.

> **NOTE**
> BitLocker drive encryption provides low-level encryption of the contents of a disk volume.

If you have multiple drives encrypted, you must disable BitLocker on any data drives before disabling BitLocker on the operating system drive.

This step runs only in a standard operating system. It does not run in Windows PE.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the action taken in this step.

Description
Provides more detailed information about the action taken in this step.

Current operating system drive
Disables BitLocker on the current operating system drive.

Specific drive
Disables BitLocker on a specific drive. Use the drop-down list to specify the drive where BitLocker is disabled.

Download Package Content

Use the **Download Package Content** task sequence step to download any of the following package types:

- Operating system images
- Operating system upgrade packages
- Driver packages
- Packages

This step works well in a task sequence to upgrade an operating system in the following scenarios :

- To use a single upgrade task sequence that can work with both x86 and x64 platforms. To do this, include two **Download Package Content** steps in the **Prepare for Upgrade** group with conditions to detect the client architecture and download only the appropriate operating system upgrade package. Configure each **Download Package Content** step to use the same variable, and use the variable for the media path on the **Upgrade Operating System** step.
- To dynamically download an applicable driver package, use two **Download Package Content** steps with conditions to detect the appropriate hardware type for each driver package. Configure each **Download Package Content** step to use the same variable, and use the variable for the **Staged content** value in drivers section on the **Upgrade Operating System** step.

This step runs in either a standard operating system or Windows PE. However, the option to save the package in the Configuration Manager client cache is not supported in WinPE.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the action taken in this step.

Description
Provides more detailed information about the action taken in this step.

Select package icon
Click the icon to select the package to download. After you select a package, you can click the icon again to choose another package.

Place into the following location
Choose to save the package in one of the following locations:

- **Task sequence working directory**
- **Configuration Manager client cache**: You use this option to store the content in the clients cache. This allows the client to act as a peer cache source for other peer cache clients. For more information, see Prepare Windows PE peer cache to reduce WAN traffic.
- **Custom path**

Save path as a variable
You can save the path as a variable that you can use in another task sequence step. Configuration Manager adds a numerical suffix to the variable name. For example, if you specify a variable of %*mycontent*% as a custom variable, it is the root for where all the referenced content is stored (which can be multiple packages). When you refer to the variable, you will add a numerical suffix to the variable. For example, for the first package, you will refer to %*mycontent01*% variable. When you refer to the variable in a subsequence steps, such as Upgrade Operating System, you would use %*mycontent02*% or %*mycontent03*% where the number corresponds to the order in which the package is listed in the step.

If a package download fails, continue downloading other packages in the list
Specifies that if the package download fails that it will go to the next package in the list and start the download.

Enable BitLocker

Use the **Enable BitLocker** task sequence step to enable BitLocker encryption on at least two partitions on the hard drive. The first active partition contains the Windows bootstrap code. Another partition contains the operating system. The bootstrap partition must remain unencrypted.

Use the **Pre-provision BitLocker** task sequence step to enable BitLocker on a drive while in Windows PE. For more information, see the Pre-provision BitLocker section in this topic.

> **NOTE**
> BitLocker drive encryption provides low-level encryption of the contents of a disk volume.

The **Enable BitLocker** step runs only in a standard operating system. It does not run in Windows PE. For more information about the task sequence variables for this action, see Enable BitLocker Task Sequence Action Variables.

The Trusted Platform Module (TPM) must be in the following state when you specify **TPM Only**, **TPM and Startup Key on USB** or **TPM and PIN**, before you can run the **Enable BitLocker** step:

- Enabled
- Activated
- Ownership Allowed

The task sequence step can complete any remaining TPM initialization, because the remaining steps do not require physical presence or reboots. The remaining TPM initialization steps which can be completed transparently by **Enable BitLocker** (if necessary) include:

- Create endorsement key pair
- Create owner authorization value and escrow to Active Directory, which must have been extended to support this value
- Take ownership
- Create the storage root key, or reset if already present but incompatible

If you want the **Enable BitLocker** step to wait until the drive encryption process has been completed before continuing with the next step in the task sequence, select the **Wait** check box. If you do not select the **Wait** check box, the drive encryption process will be performed in the background and task sequence execution will proceed immediately to the next step.

BitLocker can be used to encrypt multiple drives on a computer system (both operating system and data drives). To encrypt a data drive, the operating system must already be encrypted and the encryption process must be completed, because the key protectors for the data drives are stored on the operating system drive. As a result, if you encrypt the operating system drive and the data drive in the same process, the wait option must be selected for the step that enables BitLocker for the operating system drive.

If the hard drive is already encrypted but BitLocker is disabled then Enable BitLocker re-enables the key protector or protectors and will be completed almost instantly. Re-encryption of the hard drive is not necessary in this case.

For more information about the task sequence variables for this action, see Enable BitLocker Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name

Specifies a descriptive name for this task sequence step.

Description

Allows you to optionally enter a description for this task sequence step.

Choose the drive to encrypt

Specifies the drive to encrypt. To encrypt the current operating system drive, select **Current operating system drive** and then configure one of the following options for key management:

- **TPM only**: Select this option to use only Trusted Platform Module (TPM).
- **Startup Key on USB only**: Select this option to use a startup key stored on a USB flash drive. When you select this option, BitLocker locks the normal boot process until a USB device that contains a BitLocker startup key is attached to the computer.
- **TPM and Startup Key on USB**: Select this option to use TPM and a startup key stored on a USB flash drive. When you select this option, BitLocker locks the normal boot process until a USB device that contains a BitLocker startup key is attached to the computer.

- **TPM and PIN**: Select this option to use TPM and a personal identification number (PIN). When you select this option, BitLocker locks the normal boot process until the user provides the PIN.

To encrypt a specific, non-operating system data drive, select **Specific drive**, and then select the drive from the list.

Chose where to create the recovery key
To specify where the recovery password is created, select **In Active Directory** to escrow the password in Active Directory. If you select this option you must extend Active Directory for the site so that the associated BitLocker recovery information is saved. You can decide to not create a password at all by selecting **Do not create recovery key**. However, creating a password is a best practice.

Wait for BitLocker to complete the drive encryption process on all drives before continuing task sequence execution
Select this option to allow the BitLocker drive encryption to be completed prior to running the next step in the task sequence. If this option is selected the entire disk volume will be encrypted before the user is able to log in to the computer.

The encryption process can take hours to be completed when a large hard drive is being encrypted. Not selecting this option will allow the task sequence to proceed immediately.

Format and Partition Disk

Use the **Format and Partition Disk** task sequence step to format and partition a specified disk on the destination computer.

> **IMPORTANT**
> Every setting you specify for this task sequence step applies to a single specified disk. If you want to format and partition another disk on the destination computer, you must add an additional **Format and Partition Disk** task sequence step to the task sequence.

This task sequence step runs only in Windows PE. It does not run in a standard operating system. For more information about the task sequence variables for this action, see Format and Partition Disk Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Disk Number
The physical disk number of the disk that will be formatted. The number is based on Windows disk enumeration ordering.

Disk Type

The type of the disk that is formatted. There are two options to select from the drop-down list:

- Standard(MBR) - Master Boot Record.
- GPT - GUID Partition Table

> **NOTE**
> If you change the disk type from **Standard (MBR)** to **GPT**, and the partition layout contains an extended partition, all extended and logical partitions will be removed from the layout. You will be prompted to confirm this action before changing the disk type.

Volume
Specific information about the partition or volume that will be created, including the following:

- Name
- Remaining disk space

To create a new partition, click **New** to launch the **Partition Properties** dialog box. You can specify the partition type and size, and specify if this will be a boot partition. To modify an existing partition, click the partition to be modified and then click the properties button. For more information about how to configure hard drive partitions, see one of the following:

- How to Configure UEFI/GPT-Based Hard Drive Partitions
- How to Configure BIOS/MBR-Based Hard Drive Partitions

To delete a partition, select the partition to be deleted and then click **Delete**.

Install Application

Use the **Install Application** task sequence step to install applications as part of the task sequence. This step can install a set of applications that are specified by the task sequence step or a set of applications that are specified by a dynamic list of task sequence variables. When this step is run, the application installation begins immediately without waiting for a policy polling interval.

The applications that are installed must meet the following criteria:

- The application must be a deployment type of Windows Installer or Script installer. Windows app package (.appx file) deployment types are not supported.
- It must run under the local system account and not the user account.
- It must not interact with the desktop. The program must run silently or in an unattended mode.
- It must not initiate a restart on its own. The application must request a restart by using the standard restart code, a 3010 exit code. This ensures that the task sequence step will handle the restart correctly. If the application does return a 3010 exit code, the underlying task sequence engine performs the restart. After the restart, the task sequence automatically continues.

When the **Install Application** step runs, the application checks the applicability of the requirement rules and detection method on the deployment types of the application. Based on the results of this check, the application installs the applicable deployment type. If a deployment type contains dependencies, the dependent deployment type is evaluated and installed as part of the install application step. Application dependencies are not supported for stand-alone media.

> **NOTE**
> To install an application that supersedes another application, the content files for the superseded application must be available or the task sequence step will fail. For example, Microsoft Visio 2010 is installed on a client or in a captured image. When the Install Application task sequence step is run to install Microsoft Visio 2013, the content files for Microsoft Visio 2010 (the superseded application) must be available on a distribution point or the task sequence will fail. A client or captured image without Microsoft Visio installed will complete the Microsoft Visio 2013 installation without checking for the Microsoft Visio 2010 content files.

> **NOTE**
> You can use the SMSTSMPListRequestTimeoutEnabled and SMSTSMPListRequestTimeout built-in variables to enable and specify how many milliseconds a task sequence waits before it retries to install an application or software update after it fails to retrieve the management point list from location services. For more information, see Task sequence built-in varliables.

This task sequence step runs only in a standard operating system. It does not run in Windows PE.

Details

On the **Properties** tab for this step, you can configure the settings that are described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify to retry this step if the computer unexpectedly restarts. You can also specify how many times to retry after a restart.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Install the following applications
This setting specifies the applications that are installed in the order that they are specified.

Configuration Manager will filter out any disabled applications or any applications with the following settings. These applications will not appear in the **Select the application to install** dialog box.

- Only when a user is logged on
- Run with user rights

Install applications according to dynamic variable list
This setting specifies the base name for a set of task sequence variables that are defined for a collection or for a computer. These variables specify the applications that will be installed for that collection or computer. Each variable name consists of its common base name plus a numerical suffix starting at 01. The value for each variable must contain the name of the application and nothing else.

For applications to be installed by using a dynamic variable list, the following setting must be enabled on the **General** tab of the application's **Properties** dialog box: **Allow this application to be installed from the Install Application task sequence action instead of deploying manually**

> **NOTE**
> You cannot install applications by using a dynamic variable list for stand-alone media deployments.

For example, to install a single application by using a task sequence variable called AA01, you specify the following variable:

VARIABLE NAME	VARIABLE VALUE
AA01	Microsoft Office

To install two applications, you would specify the following variables:

VARIABLE NAME	VARIABLE VALUE
AA01	Microsoft Lync
AA02	Microsoft Office

The following conditions will affect what is installed:

- If the value of a variable contains any information other than the name of the application. That application is not installed and the task sequence continues.
- If no variable with the specified base name and "01" suffix are found, no applications are installed. When you select **Continue on error** on the Options tab of the task sequence step, the task sequence continues when an application fails to install. When the setting is not selected, the task sequence fails and will not install remaining applications.

If an application fails, continue installing other applications in the list
This setting specifies that the step continues if an individual application installation fails. If this setting is specified, the task sequence will continue regardless of any installation errors that are returned. If this is not specified an installation fails, the task sequence step will end immediately.

Install Deployment Tools

Use the **Install Deployment Tools** task sequence step to install the Configuration Manager package that contains the Sysprep deployment tools.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Sysprep Package

This setting specifies the Configuration Manager package that contains the Sysprep deployment tools for the following operating systems:

- Windows XP SP3
- Windows XP X64 SP2
- Windows Server 2003 SP2

Install Package

Use the **Install Package** task sequence step to install software as part of the task sequence. When this step is run, the installation begins immediately without waiting for a policy polling interval

The software that is installed must meet the following criteria:

- It must run under the local system account and not the user account.
- It should not interact with the desktop. The program must run silently or in an unattended mode.
- It must not initiate a restart on its own. The software must request a restart using the standard restart code, a 3010 exit code. This ensures that the task sequence step will properly handle the restart. If the software does return a 3010 exit code, the underlying task sequence engine will perform the restart. After the restart, the task sequence will automatically continue.

Programs that use the **Run another program first** option to install a dependent program are not supported when deploying an operating system. If **Run another program first** is enabled for the software and the dependent program has already been run on the destination computer, the dependent program will be run and the task sequence will continue. However, if the dependent program has not already been run on the destination computer, the task sequence step will fail.

> **NOTE**
>
> The central administration site does not have the necessary client configuration policies that are required to enable the software distribution agent during the execution of the task sequence. When you create stand-alone media for a task sequence at the central administration site, and the task sequence includes an **Install Package** step, the following error might appear in the CreateTsMedia.log file:
>
> ```
> "WMI method SMS_TaskSequencePackage.GetClientConfigPolicies failed (0x80041001)"
> ```
>
> For stand-alone media that includes an Install Package step, you must create the stand-alone media at a primary site that has the software distribution agent enabled or add a **Run Command Line** step after the **Setup Windows and ConfigMgr** step and before the first **Install Package** step. The **Run Command Line** step runs a WMIC command to enable the software distribution agent before the first Install package step runs. You can use the following in your **Run Command Line** task sequence step:
>
> **Command Line**: **WMIC /namespace:\\root\ccm\policy\machine\requestedconfig path ccm_SoftwareDistributionClientConfig CREATE ComponentName="Enable SWDist", Enabled="true", LockSettings="TRUE", PolicySource="local", PolicyVersion="1.0", SiteSettingsKey="1" /NOINTERACTIVE**
>
> For more information about creating stand-alone media, see Create stand-alone media.

This task sequence step runs only in a standard operating system. It does not run in Windows PE.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.

- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Install a single software package
This setting specifies a Configuration Manager software package. The step will wait until the installation is complete.

Install software packages according to dynamic variable list
This setting specifies the base name for a set of task sequence variables that are defined for a collection or for a computer. These variables specify the packages that will be installed for that collection or computer. Each variable name consists of its common base name plus a numerical suffix starting at 001. The value for each variable must contain a package ID and the name of the software separated by a colon.

For software to be installed by using a dynamic variable list, the following setting must be enabled on the **Advanced** tab of the package's **Properties** dialog box: **Allow this program to be installed from the Install Package task sequence without being deployed**

> **NOTE**
> You cannot install software packages by using a dynamic variable list for stand-alone media deployments.

For example, to install a single software package by using a task sequence variable called AA001, you specify the following variable:

VARIABLE NAME	VARIABLE VALUE
AA001	CEN00054:Install

To install three software packages, you would specify the following variables:

VARIABLE NAME	VARIABLE VALUE
AA001	CEN00054:Install
AA002	CEN00107:Install Silent
AA003	CEN00031:Install

The following conditions will affect what is installed:

- If the value of a variable is not created in the correct format or it does not specify a valid application ID and name, the installation of the software will fail.
- If the package Id contains lowercase characters, the installation of that software will fail.
- If no variables with the specified base name and "001" suffix are found, no packages are installed and the task sequence continues.

If installation of a software package fails, continue installing other packages in the list

This setting specifies that the step continues if an individual software package installation fails. If this setting is specified, the task sequence will continue regardless of any installation errors that are returned. If this is not specified an installation fails, the task sequence step will end immediately.

Install Software Updates

Use the **Install Software Updates** task sequence step to install software updates on the destination computer. The destination computer is not evaluated for applicable software updates until this task sequence step runs. At that time, the destination computer is evaluated for software updates like any other Configuration Manager-managed client. In particular, this step installs only the software updates that are targeted to collections of which the computer is currently a member.

> **IMPORTANT**
>
> We highly recommend that you install the latest version of the Windows Update Agent for much better performance when using the Install Software Updates task sequence step.
>
> - For Windows 7, see Knowledge base article 3161647.
> - For Windows 8, see Knowledge base article 3163023.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For information about task sequence variables for this task sequence action, see Install Software Updates Task Sequence Action Variables.

> **NOTE**
>
> You can use the SMSTSMPListRequestTimeoutEnabled and SMSTSMPListRequestTimeout built-in variables to enable and specify how many milliseconds a task sequence waits before it retries to install an application or software update after it fails to retrieve the management point list from location services. For more information, see Task sequence built-in variables.

> **NOTE**
>
> On the options tab, you can configure this task sequence to retry if the computer unexpectedly restarts. For example, a software update installation that automatically restarts the computer. Beginning in Configuration Manager 1602, you can configure the SMSTSWaitForSecondReboot variable to specify how long (in seconds) the task sequence should pause after the computer restarts when installing software updates. For more information, see Task sequence built-in variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify to retry this step if the computer unexpectedly restarts. You can also specify how many times to retry after a restart.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name

A short user-defined name that describes the action taken in this step.

Description

More detailed information about the action taken in this step.

Required for installation - Mandatory software updates only
Select this option to install all software updates flagged in Configuration Manager as mandatory for the destination computers that receive the task sequence. Mandatory software updates have administrator-defined deadlines for installation.

Available for installation - All software updates
Select this option to install all available software updates targeting the Configuration Manager collection that will receive the task sequence. All available software updates will be installed on the destination computers.

Evaluate software updates from cached scan results
Beginning in Configuration Manager version 1606, you have the option to do a full scan for software updates instead of using the cached scan results. By default, the task sequence uses cached results. You can clear the checkbox to have the client connect to the software update point to process and download the latest software updates catalog. You might chose this option when you use a task sequence to capture and build an operating system image, where you know there will be a large number of software updates, especially many that have dependencies (need to install X before Y will appear as applicable). When you clear this setting and deploy the task sequence to a large number of clients, they will all connect to the software update point at the same time. This might result in performance issues during the process and download of the catalog. In most cases, we recommend that you use the default setting.

A new task sequence variable, SMSTSSoftwareUpdateScanTimeout, was introduced in Configuration Manager version 1606 to give you the ability to control the timeout for the software updates scan during the Install software updates task sequence step. The default value is 30 minutes. For more information, see Task sequence built-in variables.

Join Domain or Workgroup

Use the **Join Domain or Workgroup** task sequence step to add the destination computer to a workgroup or domain.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For information about task sequence variables for this task sequence action, see Join Domain or Workgroup Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Join a workgroup
Select this option to have the destination computer join the specified workgroup. If the computer is currently a member of a domain, selecting this option will cause the computer to reboot.

Join a domain

Select this option to have the destination computer join the specified domain.

Optionally, enter or browse for an organizational unit (OU) in the specified domain for the computer to join. If the computer is currently a member of some other domain or a workgroup, this will cause the computer to reboot. If the computer is already a member of some other OU, Active Directory Domain Services does not allow you to change the OU and this setting is ignored.

Enter the account which has permission to join the domain
Click **Set** to enter an account and password that has permissions to join the domain. The account must be entered in the following format:

Domain\account

Prepare ConfigMgr Client for Capture

Use the **Prepare ConfigMgr Client for Capture** step to remove the Configuration Manager client or configure the client on the reference computer to prepare it for capture as part of the imaging process.

Starting in Configuration Manager version 1610, the Prepare ConfigMgr Client step completely removes the Configuration Manager client, instead of only removing key information. When the task sequence deploys the captured operating system image it will install a new Configuration Manager client each time.

Prior to Configuration Manager version 1610, this step performs the following tasks:

- Removes the client configuration properties section from the smscfg.ini file in the Windows directory. These properties include client-specific information including the Configuration Manager GUID and other client identifiers.
- Deletes all SMS or Configuration Manager machine certificates.
- Deletes the Configuration Manager client cache.
- Clears the assigned site variable for the Configuration Manager client.
- Deletes all local Configuration Manager policy.
- Removes the trusted root key for the Configuration Manager client.

This task sequence step runs only in a standard operating system. It does not run in Windows PE.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Prepare Windows for Capture

Use the **Prepare Windows for Capture** task sequence step to specify the Sysprep options to use when capturing

an operating system image on the reference computer. This task sequence action runs Sysprep and then reboots the computer into Windows PE boot image specified for the task sequence. The reference computer must not be joined to a domain for this action to be completed successfully.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For information about task sequence variables for this task sequence action, see Prepare Windows for Capture Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Automatically build mass storage driver list
Select this option to have Sysprep automatically build a list of mass storage drivers from the reference computer. This option enables the Build Mass Storage Drivers option in the sysprep.inf file on the reference computer. For more information about this setting, refer to the Sysprep documentation.

Do not reset activation flag
Select this option to prevent Sysprep from resetting the product activation flag.

Pre-provision BitLocker

Use the **Pre-provision BitLocker** task sequence step to enable BitLocker on a drive while in Windows PE. Only the used drive space is encrypted, and therefore, encryption times are much faster. You apply the key management options by using the Enable BitLocker task sequence step after the operating system installs. This step runs only in Windows PE. It does not run in a standard operating system.

> **IMPORTANT**
> To pre-provision BitLocker, you must deploy a minimum operating system of Windows 7 and TPM must be supported and enabled on the computer.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specify a short user-defined name that describes the action taken in this step.

Description
Specify detailed information about the action taken in this step.

Apply BitLocker to the specified drive
Specify the drive for which you want to enable BitLocker. Only the used space on the drive is encrypted.

Skip this step for computers that do not have a TPM or when TPM is not enabled
Select this option to skip the drive encryption when the computer hardware does not support TPM or when TPM is not enabled. For example, you can use this option when you deploy an operating system to a virtual machine.

Release State Store

Use the **Release State Store** task sequence step to notify the state migration point that the capture or restore action is complete. This step is used in conjunction with the **Request State Store**, **Capture User State**, and **Restore User State** task sequence steps to migrate user state data using a state migration point and the User State Migration Tool (USMT).

For more information about managing the user state when deploying operating systems, see Manage user state.

If you requested access to a state migration point to capture user state in the **Request State Store** task sequence step, this step notifies the state migration point that the capture process is complete and that the user state data is available to be restored. The state migration point sets the access control permissions for the captured state so that it can only be accessed (as read-only) by the restoring computer.

If you requested access to a state migration point to restore user state in the **Request State Store** task sequence step, this task sequence step notifies the state migration point that the restore process is complete. At this point, whatever retention settings you configured for the state migration point are activated.

> **IMPORTANT**
> It is a best practice to set **Continue on Error** on any task sequence steps between the **Request State Store** step and **Release State Store** step so that every **Request State Store** task sequence action has a matching **Release State Store** task sequence action.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For information about task sequence variables for this task sequence action, see Release State Store Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Request State Store

Use the **Request State Store** task sequence step to request access to a state migration point when capturing state

from a computer or restoring state to a computer.

For more information about managing the user state when deploying operating systems, see Manage user state.

You can use the **Request State Store** task sequence step in conjunction with the **Release State Store**, **Capture User State**, and **Restore User State** task sequence steps to migrate computer state using a state migration point and the User State Migration Tool (USMT).

> **NOTE**
> If you have just established a new state migration point site role (SMP), it can take up to one hour to be available for user state storage. To expedite the availability of the SMP you can adjust any state migration point property setting to trigger a site control file update.

This task sequence step runs in a standard operating system and in Windows PE for offline USMT. For information about the task sequence variables for this task sequence action, see Request State Store Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Capture state from the computer
Finds a state migration point that meets the minimum requirements as configured in the state migration point settings (maximum number of clients and minimum amount of free disk space) but it does not guarantee sufficient space is available at the time of state migration. Selecting this option will request access to the state migration point for the purpose of capturing the user state and settings from a computer.

If the Configuration Manager site has multiple state migration points enabled, this task sequence step finds a state migration point that has disk space available by querying the site's management point for a list of state migration points, and then evaluating each until it finds one that meets the minimum requirements.

Restore state from another computer
Select this option to request access to a state migration point for the purpose of restoring previously captured user state and settings to a destination computer.

If the Configuration Manager site has multiple state migration points, this task sequence step finds the state migration point that has the computer state that was stored for the destination computer.

Number of retries
The number of times that this task sequence step will try to find an appropriate state migration point before failing.

Retry delay (in seconds)
The amount of time in seconds that the task sequence step waits between retry attempts.

If computer account fails to connect to a state store, use the network access account.

Specifies that the Configuration Manager network access account credentials will be used to connect to the state migration point if the Configuration Manager client cannot access the SMP state store using the computer account. This option is less secure because other computers could use the network access account to access your stored state, but might be necessary if the destination computer is not domain joined.

Restart Computer

Use the **Restart Computer** task sequence step to restart the computer running the task sequence. After the restart, the computer will automatically continue with the next step in the task sequence.

This step can be run in either a standard operating system or Windows PE. For more information about the task sequence variables for this task sequence action, see Restart computer task sequence action variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

The boot image assigned to this task sequence
Select this option for the destination computer to use the boot image that is assigned to the task sequence. The boot image will be used to run subsequent task sequence steps that run in Windows PE.

The currently installed default operating system
Select this option for the destination computer to reboot into the installed operating system.

Notify the user before restarting
Select this option to display a notification to the user that the destination computer will be restarted. This option is selected by default.

Notification message
Enter a notification message that is displayed to the user before the destination computer is restarted.

Message display time-out
Specify the amount of time in seconds that a user will be given before the destination computer is restarted. The default amount of time is sixty (60) seconds.

Restore User State

Use the **Restore User State** task sequence step to initiate the User State Migration Tool (USMT) to restore user state and settings to the destination computer. This task sequence step is used in conjunction with the **Capture User State** task sequence step.

For more information about managing the user state when deploying operating systems, see Manage user state.

You can also use the **Restore User State** task sequence step with the **Request State Store** and **Release State Store** task sequence steps if you want to save the state settings to or restore settings from a state migration point in the Configuration Manager site. With USMT 3.0 and above, this option always decrypts the USMT state store by

using an encryption key generated and managed by Configuration Manager.

The **Restore User State** task sequence step provides control over a limited subset of the most commonly used USMT options. Additional command-line options can be specified by using the OSDMigrateAdditionalRestoreOptions task sequence variable.

> **IMPORTANT**
> If you are using the **Restore User State** task sequence step for a purpose unrelated to an operating system deployment scenario, add the Restart Computer task sequence step immediately following the **Restore User State** task sequence step.

This task sequence step runs only in a standard operating system. It does not run in Windows PE. For information about the task sequence variables for this task sequence action, see Restore User State Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the action taken in this step.

Description
Specifies more detailed information about the action taken in this step.

User state migration tool package
Enter the Configuration Manager package that contains the version of USMT for this step to use when restoring the user state and settings. This package does not require a program. When the task sequence step is run, the task sequence will use the version of USMT in the package you specify. Specify a package containing the 32-bit or x64 version of USMT depending upon the architecture of the operating system to which you are restoring the state.

Restore all captured user profiles with standard options
Restores the captured user profiles with the standard options. To customize the options that will be restored, select **Customize user profile capture**.

Customize how user profiles are restored
Allows you to customize the files that you want to restore to the destination computer. Click **Files** to specify the configuration files in the USMT package you want to use for restoring the user profiles. To add a configuration file, enter the name of the file in the **Filename** box, and then click **Add**. The configuration files that will be used for the operation are listed in the Files pane. The .xml file you specify defines which user file will be restored.

Restore local computer user profiles
Restores the local computer user (i.e. not domain user) profiles. You will need to assign new passwords to the restored local user accounts because the original local user account passwords cannot be migrated. Enter the new password in the **Password** box, and confirm the password in the **Confirm Password** box.

Continue if some files cannot be restored
Continues restoring user state and settings even if some files are unable to be restored. This option is enabled by default. If you disable this option and errors are encountered while restoring files, the task sequence step will end immediately with a failure and not all files will be restored.

Enable verbose logging
Enable this option to generate more detailed log file information. When restoring state, the log Loadstate.log is generated and stored in the task sequence log folder in the \windows\system32\ccm\logs folder by default.

Run Command Line

Use the **Run Command Line** task sequence step to run a specified command line.

This step can be run in a standard operating system or Windows PE. For information about task sequence variables for this task sequence action, see Run Command Line Task Sequence Action Variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the command line that is run.

Description
Specifies more detailed information about the command line that is run.

Command line
Specifies the command line that is run. This field is required. Including file name extensions are a best practice, for example, .vbs and .exe. Include all required settings files, command-line options, or switches.

If the file name does not have a file name extension specified, Configuration Manager tries .com, .exe, and .bat. If the file name has an extension that is not an executable, Configuration Manager tries to apply a local association. For example, if the command line is readme.gif, Configuration Manager starts the application specified on the destination computer for opening .gif files.

Examples:

setup.exe /a

cmd.exe /c copy Jan98.dat c:\sales\Jan98.dat

> **NOTE**
> Command-line actions, such as output redirection, piping, or copy, as in the preceding example, must be preceded by the **cmd.exe /c** command to run successfully.

Disable 64-bit file system redirection
By default, when running on a 64-bit operating system, the executable in the command line is located and run using the WOW64 file system redirector so that 32-bit versions of operating system executables and DLLs are found. Selecting this option disables the use of the WOW64 file system redirector so that native 64-bit versions of operating system executables and DLLs can be found. Selecting this option has no effect when running on a 32-bit operating system.

Start in
Specifies the executable folder for the program, up to 127 characters. This folder can be an absolute path on the destination computer or a path relative to the distribution point folder that contains the package. This field is

optional.

Examples:

c:\officexp

i386

> **NOTE**
>
> The **Browse** button browses the local computer for files and folders, so anything you select this way must also exist on the destination computer in the same location and with the same file and folder names.

Package
When you specify files or programs on the command line that are not already present on the destination computer, select this option to specify the Configuration Manager package that contains the appropriate files. The package does not require a program. This option is not required if the specified files exist on the destination computer.

Time-out
Specifies a value that represents how long Configuration Manager will allow the command line to run. This value can be from 1 minute to 999 minutes. The default value is 15 minutes.

This option is disabled by default.

> **IMPORTANT**
>
> If you enter a value that does not allow enough time for the Run Command Line task sequence step to complete successfully, the task sequence step will fail and the entire task sequence could fail depending on other control settings. If the time-out expires, Configuration Manager will terminate the command-line process.

Run this step as the following account
Specifies that the command line is run as a Windows user account other than the local system account.

> **NOTE**
>
> When you specify another account for this step and it occurs after an operating system installation step, the account must be added to the computer before you can run simple scripts or commands and the profile for the Windows user account must be restored to run more complex programs, such as an MSI.

Account
Specifies the Run As Windows user account for the command-line task in the task sequence to be run by this action. The command line will be run with the permissions of the specified account. Click **Set** to specify the local user or domain account.

> **IMPORTANT**
>
> If a **Run Command Line** task sequence action specifying a user account is executed while in Windows PE, the action will fail because Windows PE cannot be joined to a domain. The failure will be recorded in the smsts.log file.

Run PowerShell Script

Use the **Run PowerShell Script** task sequence step to run a specified PowerShell script.

This step can be run in a standard operating system or Windows PE. To run this step in Windows PE, PowerShell must be enabled in the boot image. You can enable Windows PowerShell (WinPE-PowerShell) from the **Optional Components** tab in the properties for the boot image. For more information about how to modify a boot image, see Manage boot images.

> **NOTE**
> PowerShell is not enabled by default on Windows Embedded operating systems.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
Specifies a short user-defined name that describes the command line that is run.

Description
Specifies more detailed information about the command line that is run.

Package
Specify the Configuration Manager package that contains the PowerShell script. One package can contain multiple PowerShell scripts.

Script name
Specifies the name of the PowerShell script to run. This field is required.

Parameters
Specifies the parameters to be passed to the Windows PowerShell script. Configure the parameters as if you were adding them to the Windows PowerShell script from a command line.

> **IMPORTANT**
> Provide parameters consumed by the script, not for the Windows PowerShell command line.
>
> The following example contains valid parameters:
>
> **-MyParameter1 MyValue1 -MyParameter2 MyValue2**
>
> The following example contains invalid parameters. The bold items are Windows PowerShell command-line parameters (-nologo and -executionpolicy unrestricted) and not consumed by the script.
>
> **-nologo-executionpolicy unrestricted-File MyScript.ps1 -MyParameter1 MyValue1 -MyParameter2 MyValue2**

PowerShell execution policy
Select the PowerShell execution policy enables you to determine which Windows PowerShell scripts (if any) will be allowed to run on the computer. Choose one of the following execution policies:

- **AllSigned**: Only scripts signed by a trusted publisher can be run.
- **Undefined**: No execution policy is defined. .
- **Bypass**: Loads all configuration files and runs all scripts. If you run an unsigned script that was downloaded

from the Internet, you are not prompted for permission before it runs.

> **IMPORTANT**
> PowerShell 1.0 does not support Undefined and Bypass execution policies.

Set Dynamic Variables

Use the **Set Dynamic Variables** task sequence step to perform the following:

1. Gather information from the computer and the environment that it is in, and then set specified task sequence variables with the information.
2. Evaluate defined rules and set task sequence variables based on the variables and values configured for rules that evaluate to true.

The task sequence automatically sets the following read-only task sequence variables:

- _SMSTSMake
- _SMSTSModel
- _SMSTSMacAddresses
- _SMSTSIPAddresses
- _SMSTSSerialNumber
- _SMSTSAssetTag
- _SMSTSUUID

This step can be run in either a standard operating system or Windows PE. For more information about task sequence variables, see Task sequence action variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name

A short user-defined name for this task sequence step.

Description

More detailed information about the action taken in this step.

Dynamic rules and variables

To set a dynamic variable to use in the task sequence, you can add a rule and then specify a value for each variable that you specify for the rule or add one or more variables to set without adding a rule. When you add a rule, you can choose from the following rule categories:

- **Computer**: Use this rule category to evaluate values for Asset tag, UUID, serial number, or mac address. You can set multiple values, and if any value is true, then the rule will evaluate to true. For example, the following rule evaluates to true if the Serial Number is 5892087 regardless of whether the MAC address equals 26-

78-13-5A-A4-22.

```
IF Serial Number = 5892087 OR MAC address = 26-78-13-5A-A4-22 THEN
```

- **Location**: Use this rule category to evaluate values for the default gateway.
- **Make and Model**: Use this rule category to evaluate values for the make and model of a computer. Both the make and model must evaluate to true for the rule to evaluate to true.

  Starting in Configuration Manager version 1610, you can specify an asterisk (**) and question mark (?) as wild cards, where ***** matches multiple characters and **?** matches a single character. For example, the string "DELL*900?" will match DELL-ABC-9001 and DELL9009.

- **Task Sequence Variable**: Use this rule category to add a task sequence variable, condition, and value to evaluate. The rule evaluates to true when the value set for the variable meets the specified condition.

You can specify one or more variables that will be set for a rule that evaluates to true or set variables without using a rule. You can select from existing variables or create a custom variable.

- **Existing task sequence variables**: Use this setting to select one or more variables from a list of existing task sequence variables. Array variables are not available to select.
- **Custom task sequence variables**: Use this setting to define a custom task sequence variable. You can also specify an existing task sequence variable. This is useful to specify an existing variable array, such as OSDAdapter, since variable arrays are not in the list of existing task sequence variables.

After you select the variables for a rule, you must provide a value for each variable. The variable is set to the specified value when the rule evaluates to true. For each variable, you can select **Secret value** to hide the value of the variable. By default, some existing variables hide values, such as the OSDCaptureAccountPassword task sequence variable.

> **IMPORTANT**
> When you import a task sequence with the Set Dynamic Variables step, and **Secret value** is selected for the value of the variable, the value is removed when you import the task sequence. As a result, you must re-enter the value for the dynamic variable after you import the task sequence.

Set Task Sequence Variable

Use the **Set Task Sequence Variable** task sequence step to set the value of a variable that is used with the task sequence.

This step can be run in either a standard operating system or Windows PE. Task sequence variables are read by task sequence actions and specify the behavior of those actions. For more information about specific task sequence variables, see Task sequence action variables.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name

A short user-defined name for this task sequence step.

Description
More detailed information about the action taken in this step.

Task sequence variable
A user-defined name for the task sequence variable.

Value
The value that is associated with the task sequence variable. The value could be another task sequence variable in %% syntax.

Setup Windows and ConfigMgr

Use the **Setup Windows and ConfigMgr** task sequence step to perform the transition from Windows PE to the new operating system. This task sequence step is a required part of any operating system deployment. It installs the Configuration Manager client into the new operating system and prepares for the task sequence to continue execution in the new operating system.

This step runs only in Windows PE. It does not run in a standard operating system. For more information about task sequence variables for this task sequence action, see Setup Windows and ConfigMgr task sequence action variables.

The **Setup Windows and ConfigMgr** task sequence action replaces sysprep.inf or unattend.xml directory variables, such as %WINDIR% and %ProgramFiles%, with the Windows PE installation directory X:\Windows. Task sequence variables specified by using these environment variables will be ignored.

Use this task sequence step to perform the following actions:

1. Preliminaries: Windows PE
    a. Performs task sequence variable substitution in the unattend.xml file.
    b. Downloads the package that contains the Configuration Manager client and puts it in the deployed image.
2. Set up Windows
    a. Image-based installation.
        a. Disables the Configuration Manager client in the image (that is, disables Autostart for the Configuration Manager client service).
        b. Updates the registry in the deployed image to ensure that the deployed operating system starts with the same drive letter that it had on the reference computer.
        c. Restarts in the deployed operating system.
        d. Windows mini-setup runs by using the previously specified sysprep.inf or unattend.xml file that has all end-user interaction suppressed. Note: If **Apply Network Settings** specified to join a domain, then that information is in the sysprep.inf or unattend.xml file, and Windows mini-setup performs the domain join.
    b. Setup.exe-based installation. Runs Setup.exe that follows the typical Windows setup process:
        a. Copies the operating system install package specified in an earlier **Apply Operating System** task sequence to the hard disk drive.
        b. Restarts in the newly deployed operating system.
        c. Windows mini-setup runs by using the previously specified sysprep.inf or unattend.xml file

that has all user interface settings suppressed. Note: If **Apply Network Settings** specified to join a domain, then that information is in the sysprep.inf or unattend.xml file, and Windows mini-setup performs the domain join.

3. Set up the Configuration Manager client

    a. After Windows mini-setup finishes, the task sequence resumes by using setupcomplete.cmd.

    b. Enables or disables the local administrator account, based on the option selected in the **Apply Windows Settings** step.

    c. Installs the Configuration Manager client by using the previously downloaded package (1.b) and installation properties specified in the Task Sequence Editor. The client is installed in "provisioning mode" to prevent it from processing new policy requests until the task sequence is completed.

    d. Waits for the client to be fully operational.

    e. If the computer is operating in an environment with Network Access Protection enabled, the client checks for and installs any required updates so that all required updates are present before the task sequence continues running.

4. The task sequence continues running with its next step.

> **NOTE**
> The **Setup Windows and ConfigMgr** task sequence action is responsible for running Group Policy on the newly installed computer. The Group Policy is applied after the task sequence is finished.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Do not select to have the task sequence continue if an error occurs while running the step. If there is an error, the task sequence will fail whether or not you select this setting.
- Specify conditions that must be met for the step to run.

Name

Specifies a short user-defined name that describes the action taken in this step.

Description

Specifies additional information about the action taken in this step.

Client package

Specifies the Configuration Manager client installation package that will be used by this task sequence step. Click **Browse** and select the client installation package that you want to use to install the Configuration Manager client.

Use pre-production client package when available

Specifies that if there is a pre-production client package available, that the task sequence step will use this package instead of the production client package. Typically, the pre-production client is a newer version that is being tested in the production environment. Click **Browse** and select the pre-production client installation package that you want to use to install the Configuration Manager client.

Installation Properties

Site assignment and the default configuration are automatically specified by the task sequence action. You can use this field to specify any additional installation properties to use when you install the client. To enter multiple

installation properties, separate them with a space.

You can specify command-line options to use during client installation. For example, you can enter **/skipprereq: silverlight.exe** to inform CCMSetup.exe not to install the Microsoft Silverlight prerequisite. For more information about available command-line options for CCMSetup.exe, see About client installation properties.

Upgrade Operating System

Use the **Upgrade Operating System** task sequence step to upgrade an existing Windows 7, Windows 8, Windows 8.1, or Windows 10 operating system to a Windows 10.

This task sequence step runs only in a standard operating system. It does not run in Windows PE.

Details

On the **Properties** tab for this step, you can configure the settings described in this section.

In addition, use the **Options** tab to do the following actions:

- Disable the step.
- Specify if the task sequence continues if an error occurs while running the step.
- Specify conditions that must be met for the step to run.

Name
A short user-defined name that describes the action taken in this step.

Description
More detailed information about the action taken in this step.

Upgrade package
Select this option to specify the Windows 10 operating system upgrade package to use for the upgrade.

Source path
Specifies a local or network path to the Windows 10 media that is to be used (corresponds to the /installFrom command-line option). You can also specify a variable, such as %mycontentpath% or %DPC01%. When you use a variable for the source path, it must be specified earlier in the task sequence. For example, if you use the Download Package Content step in the task sequence, you can specify a variable for the location of the operating system upgrade package. Then, you can use that variable for the source path for this step.

Edition
Specify the edition within the operating system media to use for the upgrade.

Product key
Specify the product key to apply to the upgrade process

Provide the following driver content to Windows Setup during upgrade
Select this setting to add drivers to the destination computer during the upgrade process (corresponds to the /InstallDriver command-line option). The drivers must be compatible with Windows 10. Specify one of the following:

- **Driver package**: Click **Browse** and select an existing driver package from the list.
- **Staged content**: Select this option to specify the location for the driver package. You can specify a local folder, network path, or a task sequence variable. When you use a variable for the source path, it must be specified earlier in the task sequence. For example, by using the Download Package Content step.

Time-out (minutes)
Specifies the number of minutes Setup has to run before Configuration Manager will fail the task sequence step.

Perform Windows Setup compatibility scan without starting upgrade
Specifies to perform the Windows Setup compatibility scan without starting the upgrade process (corresponds to the /Compat ScanOnly command-line option). You must still deploy the entire installation source when you use this option. Setup returns an exit code as a result of the scan. The following table provides some of the more common exit codes.

EXIT CODE	DETAILS
MOSETUP_E_COMPAT_SCANONLY (0xC1900210)	No compatibility issues ("success").
MOSETUP_E_COMPAT_INSTALLREQ_BLOCK (0xC1900208)	Actionable compatibility issues.
MOSETUP_E_COMPAT_MIGCHOICE_BLOCK (0xC1900204)	Selected migration choice is not available. For example, an upgrade from Enterprise to Professional.
MOSETUP_E_COMPAT_SYSREQ_BLOCK (0xC1900200)	Not eligible for Windows 10.
MOSETUP_E_COMPAT_INSTALLDISKSPACE_BLOCK (0xC190020E)	Not enough free disk space.

For more information about this parameter, see Windows Setup Command-Line Options

Ignore any dismissible compatibility messages
Specifies that Setup completes the installation, ignoring any dismissible compatibility messages (corresponds to the /Compat IgnoreWarning command-line option).

Dynamically update Windows Setup with Windows Update
Specifies whether setup will perform Dynamic Update operations, such as search, download, and install updates (corresponds to the /DynamicUpdate command-line option). This setting is not compatible with Configuration Manager software updates, but it can be enabled when you handle updates by using WSUS (stand-alone) or Windows Update.

Override policy and use default Microsoft Update: Select this setting to temporarily override the local policy in realtime to run Dynamic Update operations and have the computer get updates from Windows Update.

Task sequence action variables in System Center Configuration Manager

11/23/2016 • 28 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Task sequence action variables specify configuration settings that are used by a single step in a System Center Configuration Manager task sequence. By default, the settings used by a task sequence step are initialized before the step is run and available only while the associated task sequence step is run. In other words, the task sequence variable setting is added to the task sequence environment before the task sequence step is run, and the value is removed from the task sequence environment after the task sequence step has run.

Action Variable Example

For example, you can specify a start-in directory for a command-line action by using the **Run Command Line** task sequence step. This step includes a **Start In** property whose default value is stored in the task sequence environment as the **WorkingDirectory** variable. The **WorkingDirectory** environment variable is initialized before the **Run Command Line** task sequence action is run. During the **Run Command Line** step, the **WorkingDirectory** value can be accessed through the **Start In** property. Then after the task sequence step is completed, the value of the **WorkingDirectory** variable is removed from the task sequence environment. If the sequence contains another **Run Command Line** task sequence step, the new **WorkingDirectory** variable is initialized and set to the starting value for that task sequence step.

Whereas the default value for a task sequence action setting is present while the task sequence step is run, any new value that you set can be used by multiple steps in the sequence. If you use one of the task sequence variable creation methods to override a built-in variable value, the new value remains in the environment and overrides the default value for other steps in the task sequence. In the previous example, if a **Set Task Sequence Variable** step is added as the first step of the task sequence and sets the **WorkingDirectory** environment variable to the value **C:**, both **Run Command Line** steps in the task sequence will use the new starting directory value.

Action Variables for Task Sequence Actions

Configuration Manager task sequence variables are grouped by their associated task sequence action. Use the following links to gather information about the action variables associated with a specific action. The task sequence variables govern how the task sequence action operates. The task sequence action reads and uses the variables that you mark as input variables. Alternatively, you can use the Set Task Sequence Variable action or the TSEnvironment COM object to set the variables at runtime. Only the task sequence action marks variables as output variables, which are read by actions that occur later in the task sequence.

> **NOTE**
> Not all task sequence actions are associated with a set of task sequence variables. For example, although there are variables associated with the Enable BitLocker action, there are no variables associated with the Disable BitLocker action.

Apply Data Image Task Sequence Action Variables

The variables for this action specify which image of a WIM file is applied to the destination computer and whether to delete the files on the destination partition. For more information about the task sequence step associated with these variables, see Apply Data Image Task Sequence Step.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDDataImageIndex (input)	Specifies the index value of the image that is applied to the destination computer.
OSDWipeDestinationPartition (input)	Specifies whether to delete the files located on the destination partition. Valid values: **"true"** (default) **"false"**

Apply Driver Package Task Sequence Action Variables

The variables for this action specify information the installation of mass storage drivers and whether to install unsigned drivers. For more information about the task sequence step associated with these variables, see Apply Driver Package.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDApplyDriverBootCriticalContentUniqueID (input)	Specifies the content ID of the mass storage device driver to install from the driver package. If this is not specified, no mass storage driver is installed.
OSDApplyDriverBootCriticalINFFile (input)	Specifies the INF file of the mass storage driver to install. This task sequence variable is required if the OSDApplyDriverBootCriticalContentUniqueID is set.
OSDApplyDriverBootCriticalHardwareComponent (input)	Specifies whether a mass storage device driver is installed, this must be **scsi**. This task sequence variable is required if the OSDApplyDriverBootCriticalContentUniqueID is set.
OSDApplyDriverBootCriticalID (input)	Specifies the boot critical ID of the mass storage device driver to install. This ID is listed in the "**scsi**" section of the device driver's txtsetup.oem file. This task sequence variable is required if the OSDApplyDriverBootCriticalContentUniqueID is set.

ACTION VARIABLE NAME	DESCRIPTION
OSDAllowUnsignedDriver (input)	Specifies whether to configure Windows to allow the installation of unsigned device drivers. This task sequence variable is not used when deploying the Windows Vista and later operating system. Valid values: **"true"** **"false"** (default)

Apply Network Settings Task Sequence Action Variables

The variables for this action specify network settings for the destination computer, such as settings for the network adapters of the computer, domain settings and workgroup settings. For more information about the task sequence step associated with these variables, see Apply Network Settings Step.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDAdapter (input)	This task sequence variable is an array variable. Each element in the array represents the settings for a single network adapter on the computer. The settings defined for each adapter are accessed by combining the array variable name with the zero-based network adapter index and the property name. If multiple network adapters will be configured with this task sequence action, the properties for the second network adapter are defined by using their index in the variable name; for example, OSDAdapter1EnableDHCP, OSDAdapter1IPAddressList, OSDAdapter1DNSDomain, OSDAdapter1WINSServerList, OSDAdapter1EnableWINS, and so on. For example, the following variable names can be used to define the properties for the first network adapter that will be configured by this task sequence action: • **OSDAdapter0EnableDHCP** - true to enable Dynamic Host Configuration Protocol (DHCP) for the adapter. This setting is required. Possible values are: True or False. • **OSDAdapter0IPAddressList** - Comma-delimited list of IP addresses for the adapter. This property is ignored unless **EnableDHCP** is set to **false**. This setting is required. • **OSDAdapter0SubnetMask** - Comma-delimited list of subnet masks. This property is ignored unless **EnableDHCP** is set to **false**. This setting is required. • **OSDAdapter0Gateways** - Comma-delimited list of IP gateway addresses. This property is ignored unless **EnableDHCP** is set to **false**. This setting is required. • **OSDAdapter0DNSDomain** - Domain Name System

ACTION VARIABLE NAME	DESCRIPTION
	(DNS) domain for the adapter. • **OSDAdapter0DNSServerList** - Comma-delimited list of DNS servers for the adapter. This setting is required. • **OSDAdapter0EnableDNSRegistration** - **true** to register the IP address for the adapter in DNS. • **OSDAdapter0EnableFullDNSRegistration** - **true** to register the IP address for the adapter in DNS under the full DNS name for the computer. • **OSDAdapter0EnableIPProtocolFiltering** - **true** to enable IP protocol filtering on the adapter. • **OSDAdapter0IPProtocolFilterList** - Comma-delimited list of protocols allowed to run over IP. This property is ignored if **EnableIPProtocolFiltering** is set to **false**. • **OSDAdapter0EnableTCPFiltering** - **true** to enable TCP port filtering for the adapter. • **OSDAdapter0TCPFilterPortList** - Comma-delimited list of ports to be granted access permissions for TCP. This property is ignored if **EnableTCPFiltering** is set to **false**. • **OSDAdapter0TcpipNetbiosOptions** - Options for NetBIOS over TCP/IP. Possible values are as follows: ◦ 0 Use NetBIOS settings from DHCP server. ◦ 1 Enable NetBIOS over TCP/IP. ◦ 2 Disable NetBIOS over TCP/IP. • **OSDAdapter0EnableWINS** - **true** to use WINS for name resolution. • **OSDAdapter0WINSServerList** - Comma-delimited list of WINS server IP addresses. This property is ignored unless **EnableWINS** is set to **true**. • **OSDAdapter0MacAddress** - Media access controller (MAC) address used to match settings to physical network adapter. • **OSDAdapter0Name** - Name of the network connection as it appears in the network connections control panel program. The name is between 0 and 255 characters in length. • **OSDAdapter0Index** - Index of the network adapter settings in the array of settings. OSDAdapterCount=1 OSDAdapter0EnableDHCP=FALSE OSDAdapter0IPAddressList=192.168.0.40 OSDAdapter0SubnetMask=255.255.255.0 OSDAdapter0Gateways=192.168.0.1 OSDAdapter0DNSSuffix=contoso.com
OSDAdapterCount (input)	Specifies the number of network adapters installed on the destination computer. When the **OSDAdapterCount** value is set, all the configuration options for each adapter must be set. For example, if you set the **OSDAdapterTCPIPNetbiosOptions** value for a specific adapter then all the values for that adapter must also be configured. If this value is not specified, all **OSDAdapter** values are ignored.

ACTION VARIABLE NAME	DESCRIPTION
OSDDNSDomain (input)	Specifies the primary DNS server that is used by the destination computer.
OSDDomainName (input)	Specifies the name of the Windows domain that the destination computer joins. The specified value must be a valid Active Directory Domain Services domain name.
OSDDomainOUName (input)	Specifies the RFC 1779 format name of the organizational unit (OU) that the destination computer joins. If specified, the value must contain the full path. Example: **LDAP://OU=MyOu,DC=MyDom,DC=MyCompany,DC=com**
OSDEnableTCPIPFiltering (input)	Specifies whether TCP/IP filtering is enabled. Valid values: **"true"** **"false"** (default)
OSDJoinAccount (input)	Specifies the network account that is used to add the destination computer to a Windows domain.
OSDJoinPassword (input)	Specifies the network password that is used to add the destination computer to a Windows domain.
OSDNetworkJoinType (input)	Specifies whether the destination computer joins a Windows domain or a workgroup. **"0"** indicates that the destination computer joins a Windows domain. **"1"** specifies that the computer joins a workgroup. Valid values: **"0"** **"1"**
OSDDNSSuffixSearchOrder (input)	Specifies the DNS search order for the destination computer.

ACTION VARIABLE NAME	DESCRIPTION
OSDWorkgroupName (input)	Specifies the name of the workgroup that the destination computer joins. You must specify either this value or the **OSDDomainName** value. The workgroup name can be a maximum of 32 characters. Example: **"Accounting"**

Apply Operating System Image Task Sequence Action Variables

The variables for this action specify settings for the operating system that you want to install on the destination computer. For more information about the task sequence step associated with these variables, see Apply Operating System Image.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDConfigFileName (input)	Specifies the file name of the operating system deployment answer file associated with the operating system deployment package.
OSDImageIndex (input)	Specifies the image index value of the WIM file that is applied to the destination computer.
OSDInstallEditionIndex (input)	Specifies the version of Windows Vista or later operating system that is installed. If no version is specified, Windows setup will determine which version to install using the referenced product key. Use only a value of zero (0) if the following conditions are true: - You are installing a pre-Windows Vista operating system - You are installing a volume license edition of Windows Vista or later, and no product key is specified. Valid values: **"0"** (default)
OSDTargetSystemDrive (output)	Specifies the drive letter of the partition that contains the operating system files.

Apply Windows Settings Task Sequence Action Variables

The variables for this action specify Windows settings for the destination computer, such as the computer name, Windows product key, registered user and organization, and the local administrator password. For more information about the task sequence step associated with these variables, see Apply Windows Settings.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDComputerName (input)	Specifies the name of the destination computer. Example: **"%_SMSTSMachineName%"** (default)
OSDProductKey (input)	Specifies the Windows product key. The specified value must be between 1 and 255 characters.
OSDRegisteredUserName (input)	Specifies the default registered user name in the new operating system. The specified value must be between 1 and 255 characters.
OSDRegisteredOrgName (input)	Specifies the default registered organization name in the new operating system. The specified value must be between 1 and 255 characters.
OSDTimeZone (input)	Specifies the default time zone setting that is used in the new operating system.
OSDServerLicenseMode (input)	Specifies the Windows Server license mode that is used. Valid values: **"PerSeat"** **"PerServer"**
OSDServerLicenseConnectionLimit (input)	Specifies the maximum number of connections allowed. The specified number must be in the range between 5 and 9999 connections.
OSDRandomAdminPassword (input)	Specifies a randomly generated password for the administrator account in the new operating system. If set to **true**, the local administrator account will be disabled on the target computer. If set to **false**, the local administrator account will be enabled on the target computer, and the local administrator account password will be assigned the value of the variable **OSDLocalAdminPassword**. Valid values: **"true"** (default) **"false"**
OSDLocalAdminPassword (input)	Specifies the local administrator password. This value is ignored if the **Randomly generate the local administrator password and disable the account on all supported platforms** option is enabled. The specified value must be between 1 and 255 characters.

Auto Apply Drivers Task Sequence Action Variables

The variables for this action specify which Windows drivers are installed on the destination computer and whether unsigned drivers are installed. For more information about the task sequence step associated with these variables,

see Auto Apply Drivers.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDAutoApplyDriverCategoryList (input)	A comma-delimited list of the driver catalog category unique IDs. If specified, the **Auto Apply Driver** task sequence action considers only those drivers that are in at least one of these categories when installing drivers. This value is optional, and it is not set by default. The available category IDs can be obtained by enumerating the list of **SMS_CategoryInstance** objects on the site.
OSDAllowUnsignedDriver (input)	Specifies whether Windows is configured to allow unsigned device drivers to be installed. This task sequence variable is not used when deploying Windows Vista and later operating systems. Valid values: **"true"** **"false"** (default)
OSDAutoApplyDriverBestMatch (input)	Specifies what the task sequence action does if there are multiple device drivers in the driver catalog that are compatible with a hardware device. If set to **"true"**, only the best device driver will be installed. If **false**, all compatible device drivers will be installed, and the operating system will choose the best driver to use. Valid values: **"true"** (default) **"false"**

Capture Network Settings Task Sequence Action Variables

The variables for this action specify whether the network adapter settings (TCP/IP, DNS, and WINS) configuration information is captured and whether the workgroup or domain membership information is migrated as part of the operating system deployment. For more information about the task sequence step associated with these variables, see Capture Network Settings.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateAdapterSettings (input)	Specifies whether the network adapter settings (TCP/IP, DNS, and WINS) configuration information is captured. Examples: **"true"** (default) **"false"**

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateNetworkMembership (input)	Specifies whether the workgroup or domain membership information is migrated as part of the operating system deployment. Examples: **"true"** (default) **"false"**

Capture Operating System Image Task Sequence Action Variables

The variables for this action specify information about the operating system image that is being captured, such as where the image is stored, who created the image, and a description of the image. For more information about the task sequence step associated with these variables, see Capture Operating System Image.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDCaptureAccount (input)	Specifies a Windows account name that has permissions to store the captured image on a network share.
OSDCaptureAccountPassword (input)	Specifies the password for the Windows account used to store the captured image on a network share.
OSDCaptureDestination (input)	Specifies the location where the captured operating system image is saved. The maximum directory name length is 255 characters.
OSDImageCreator (input)	An optional name of the user who created the image. This name is stored in the WIM file. The maximum length of the user name is 255 characters.
OSDImageDescription (input)	An optional user-defined description of the captured operating system image. This description is stored in the WIM file. The maximum length of the description is 255 characters.
OSDImageVersion (input)	An optional user-defined version number to assign to the captured operating system image. This version number is stored in the WIM file. This value can be any combination of letters with a maximum length of 32 characters.
OSDTargetSystemRoot (input)	Specifies the path to the Windows directory of the installed operating system on the reference computer. This operating system is verified as being a supported operating system for capture by Configuration Manager.

Capture User State Task Sequence Action Variables

The variables for this action specify information used by the User State Migration Tool (USMT), such as the folder where the user state is saved, command-line options for USMT, and the configuration files used to control the capture of the user profiles. For more information about the task sequence step associated with these variables, see Capture User State.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDStateStorePath (input)	The UNC or local path name of the folder where the user state is saved. No default.
OSDMigrateAdditionalCaptureOptions (input)	Specifies user state migration tool (USMT) command line options that are used when capturing the user state, but not exposed in the Configuration Manager user interface. The additional options are specified in the form of a string that is appended to the automatically generated USMT command line. The USMT options specified with this task sequence variable are not validated for accuracy prior to running the task sequence.
OSDMigrateMode (input)	Allows you to customize the files that are captured by USMT. If this variable is set to "Simple," then only the standard USMT configuration files are used. If this variable is set to "Advanced," then the task sequence variable OSDMigrateConfigFiles specifies the configuration files that the USMT uses. Valid values: **"Simple"** **"Advanced"**
OSDMigrateConfigFiles (input)	Specifies the configuration files used to control the capture of user profiles. This variable is used only if OSDMigrateMode is set to "Advanced". This comma-delimited list value is set to perform customized user profile migration. Example: miguser.xml,migsys.xml,migapps.xml
OSDMigrateContinueOnLockedFiles (input)	Allows the user state capture to proceed if some files cannot be captured. Valid values: **"true"** (default) **"false"**
OSDMigrateEnableVerboseLogging (input)	Enables verbose logging for the USMT. Valid values: **"true"** **"false"** (default)

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateSkipEncryptedFiles (input)	Specifies whether encrypted files are captured. Valid values: **"true"** **"false"** (default)
_OSDMigrateUsmtPackageID (input)	Specifies the package ID of the Configuration Manager package that will contain the USMT files. This variable is required.

Capture Windows Settings Task Sequence Action Variables

The variables for this action specify whether specific Windows settings are migrated to the destination computer, such as the name of the computer, the register organization name, and time zone information. For more information about the task sequence step associated with these variables, see Capture Windows Settings.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateComputerName (input)	Specifies whether the computer name is migrated. Valid values: **"true"** (default) **"false"** If the value is "true," then the OSDComputerName variable is set to the NetBIOS name of the computer.
OSDComputerName (output)	Set to the NetBIOS name of the computer. The value is set only if the OSDMigrateComputerName variable is set to "true."
OSDMigrateRegistrationInfo (input)	Specifies whether the computer user and organizational information is migrated. Valid values: **"true"** (default) **"false"** If the value is "true," then the OSDRegisteredOrgName variable is set to the registered organization name of the computer.
OSDRegisteredOrgName (output)	Set to the registered organization name of the computer. The value is set only if the OSDMigrateRegistrationInfo variable is set to "true."

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateTimeZone (input)	Specifies whether the computer time zone is migrated. Valid values: **"true"** (default) **"false"** If the value is "true," then the variable OSDTimeZone is set to the time zone of the computer.
OSDTimeZone (output)	Set to the time zone of the computer. The value is set only if the OSDMigrateTimeZone variable is set to "true."

Connect to Network Folder Task Sequence Action Variables

The variables for this action specify information about a folder on a network, such as the account used and password to connect to the network folder, the drive letter of the folder, and the path to the folder. For more information about the task sequence step associated with these variables, see Connect To Network Folder.

Details

ACTION VARIABLE NAME	DESCRIPTION
SMSConnectNetworkFolderAccount (input)	Specifies the administrator account that is used to connect to the network share.
SMSConnectNetworkFolderDriveLetter (input)	Specifies the network drive letter to connect to. This value is optional; if it is not specified, then the network connection is not mapped to a drive letter. If this value is specified, the value must be in the range from D: to Z:. In addition, do not use X: as it is the drive letter used by Windows PE during the Windows PE phase. Examples: **"D:"** **"E:"**
SMSConnectNetworkFolderPassword (input)	Specifies the network password that is used to connect to the network share.
SMSConnectNetworkFolderPath (input)	Specifies the network path for the connection. Example: **"\\servername\sharename"**

Convert Disk to Dynamic Task Sequence Action Variables

The variable for this action specifies the number of the physical disk to convert from a basic to dynamic disk. For more information about the task sequence step associated with these variables, see Convert Disk to Dynamic.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDConvertDiskIndex (input)	Specifies the physical disk number that is converted.

Enable BitLocker Task Sequence Action Variables

The variables for this action specify the recovery password and startup key options used to enable BitLocker on the destination computer. For more information about the task sequence step associated with these variables, see Enable BitLocker.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDBitLockerRecoveryPassword (input)	Instead of generating a random recovery password, the **Enable BitLocker** task sequence action uses the specified value as the recovery password. The value must be a valid numerical BitLocker recovery password.
OSDBitLockerStartupKey (input)	Instead of generating a random startup key for the key management option **Startup Key on USB only,** the **Enable BitLocker** task sequence action uses the Trusted Platform Module (TPM) as the startup key. The value must be a valid, 256-bit Base64-encoded BitLocker startup key.

Format and Partition Disk Task Sequence Action Variables

The variables for this action specify information for formatting and partitioning a physical disk, such as the disk number and an array of partition settings. For more information about the task sequence step associated with these variables, see Format and Partition Disk.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDDiskIndex (input)	Specifies the physical disk number to be partitioned.
OSDDiskpartBiosCompatibilityMode (input)	Specifies whether to disable cache alignment optimizations when partitioning the hard disk for compatibility with certain types of BIOS. This can be necessary when deploying Windows XP or Windows Server 2003 operating systems. For more information, see article 931760 and article 931761 in the Microsoft Knowledge Base. Valid values: **"true"** **"false"** (default)

ACTION VARIABLE NAME	DESCRIPTION
OSDGPTBootDisk (input)	Specifies whether to create an EFI partition on a GPT hard disk so that it can be used as the startup disk on EFI-based computers. Valid values: **"true"** **"false"** (default)

ACTION VARIABLE NAME	DESCRIPTION
OSDPartitions (input)	Specifies an array of partition settings; see the SDK topic for accessing array variables in the task sequence environment. This task sequence variable is an array variable. Each element in the array represents the settings for a single partition on the hard disk. The settings defined for each partition can be accessed by combining the array variable name with the zero-based disk partition number and the property name. For example, the following variable names can be used to define the properties for the first partition that will be created by this task sequence action on the hard disk: - **OSDPartitions0Type** - Specifies the type of partition. This is a required property. Valid values are "**Primary**", "**Extended**", "**Logical**", and "**Hidden**". - **OSDPartitions0FileSystem** - Specifies the type of file system to use when formatting the partition. This is an optional property; if no file system is specified, the partition will not be formatted. Valid values are "**FAT32**" and "**NTFS**". - **OSDPartitions0Bootable** - Specifies whether the partition is bootable. This is a required property. If this value is set to "**TRUE**" for MBR disks, then this will be made the active partition. - **OSDPartitions0QuickFormat** - Specifies the type of format that is used. This is a required property. If this value is set to "**TRUE**", a quick format will be performed; otherwise, a full format will be performed. - **OSDPartitions0VolumeName** - Specifies the name that is assigned to the volume when it is formatted. This is an optional property. - **OSDPartitions0Size** - Specifies the size of the partition. Units are specified by the **OSDPartitions0SizeUnits** variable. This is an optional property. If this property is not specified, the partition is created using all remaining free space. - **OSDPartitions0SizeUnits** - Specifies the units that will be used when interpreting the **OSDPartitions0Size** task sequence variable. This is an optional property. Valid values are "**MB**" (default), "**GB**", and "**Percent**". - **OSDPartitions0VolumeLetterVariable** - Partitions will always use the next available drive letter in Windows PE when they are created. Use this optional property to specify the name of another task sequence variable, which will be used to save the new drive letter for future reference. If multiple partitions will be defined with this task sequence action, the properties for the second partition can be defined by using their index in the variable name; for example, **OSDPartitions1Type**, **OSDPartitions1FileSystem**, **OSDPartitions1Bootable**, **OSDPartitions1QuickFormat**, **OSDPartitions1VolumeName,** and so on.

ACTION VARIABLE NAME	DESCRIPTION
OSDPartitionStyle (input)	Specifies the partition style to use when partitioning the disk. "**MBR**" indicates the master boot record partition style, and "**GPT**" indicates the GUID Partition Table style. Valid Values: **"GPT"** **"MBR"**

Install Software Updates Task Sequence Action Variables

The variable for this action specifies whether to install all updates or only mandatory updates. For more information about the task sequence step associated with these variables, see Install Software Updates.

Details

ACTION VARIABLE NAME (INPUT)	DESCRIPTION
SMSInstallUpdateTarget (input)	Specifies whether to install all updates or only mandatory updates. Valid values: **"All"** **"Mandatory"**

Join Domain or Workgroup Task Sequence Action Variables

The variables for this action specify information needed to join the destination computer to a Windows domain or workgroup. For more information about the task sequence step associated with these variables, see Join Domain or Workgroup.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDJoinAccount (input)	Specifies the account that is used by the destination computer to join the Windows domain. This variable is required when joining a domain.
OSDJoinDomainName (input)	Specifies the name of a Windows domain the destination computer joins. The length of the Windows domain name must be between 1 and 255 characters.
OSDJoinDomainOUName (input)	Specifies the RFC 1779 format name of the organizational unit (OU) that the destination computer joins. If specified, the value must contain the full path. The length of the Windows domain OU name must be between 0 and 32,767 characters. This value is not set if the **OSDJoinType** variable is set to "1" (join workgroup). Example: **LDAP://OU=MyOu,DC=MyDom,DC=MyCompany,DC=com**

ACTION VARIABLE NAME	DESCRIPTION
OSDJoinPassword (input)	Specifies the network password that is used by the destination computer to join the Windows domain. If the variable is not specified then a blank password is tried. This value is required if the variable **OSDJoinType** variable is set to "**0**" (join domain).
OSDJoinSkipReboot (input)	Specifies whether to skip restarting after the destination computer joins the domain or workgroup. Valid values: **"true"** **"false"**
OSDJoinType (input)	Specifies whether the destination computer joins a Windows domain or a workgroup. To join the destination computer to a Windows domain specify "**0**". To join the destination computer to a workgroup specify "**1**". Valid values: **"0"** **"1"**
OSDJoinWorkgroupName (input)	Specifies the name of a workgroup that the destination computer joins. The length of the workgroup name must be between 1 and 32 characters. Example: **"Accounting"**

Prepare Windows for Capture Task Sequence Action Variables

The variables for this action specify information used to capture the Windows operating system from the target computer. For more information about the task sequence step associated with these variables, see Prepare ConfigMgr Client for Capture.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDBuildStorageDriverList (input)	Specifies whether sysprep builds a mass storage device driver list. This setting applies to only Windows XP and Windows Server 2003. It will populate the [SysprepMassStorage] section of sysprep.inf with information on all the mass storage drivers that are supported by the image to be captured. Valid values: **"true"** **"false"** (default)

ACTION VARIABLE NAME	DESCRIPTION
OSDKeepActivation (input)	Specifies whether sysprep resets the product activation flag. Valid values: **"true"** **"false"** (default)
OSDTargetSystemRoot (output)	Specifies the path to the Windows directory of the installed operating system on the reference computer. This operating system is verified as being a supported operating system for capture by Configuration Manager.

Release State Store Sequence Action Variables

The variables for this action specify information used to release the stored user state. For more information about the task sequence step associated with these variables, see Release State Store.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDStateStorePath (input)	The UNC or local pathname to the location from which the user state is restored. This value is used by both the **Capture User State** task sequence action and the **Restore User State** task sequence action.

Request State Store Task Sequence Action Variables

The variables for this action specify information used to request the stored user state, such as the folder on the state migration point where the user data is stored. For more information about the task sequence step associated with these variables, see Release State Store.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDStateFallbackToNAA (input)	Specifies whether the Network Access Account is used as a fallback when the computer account fails to connect to the state migration point. Valid values: **"true"** **"false"** (default)
OSDStateSMPRetryCount (input)	Specifies the number of times that the task sequence step tries to find a state migration point before the step fails. The specified count must be between 0 and 600.
OSDStateSMPRetryTime (input)	Specifies the number of seconds that the task sequence step waits between retry attempts. The number of seconds can be a maximum of 30 characters.
OSDStateStorePath (output)	The UNC path to the folder on the state migration point where the user state is stored.

Restart Computer Task Sequence Action Variables

The variables for this action specify information used to restart the destination computer. For more information about the task sequence step associated with these variables, see Restart Computer.

Details

ACTION VARIABLE NAME	DESCRIPTION
SMSRebootMessage (input)	Specifies the message to be displayed to users before restarting the destination computer. If this variable is not set, the default message text is displayed. The specified message must not exceed 512 characters. Example: - "This computer will be restarted; please save your work."
SMSRebootTimeout (input)	Specifies the number of seconds that the warning is displayed to the user before the computer restarts. Specify zero seconds to indicate that no reboot message is displayed. Examples: **"0"** (default) **"5"** **"10"**

Restore User State Task Sequence Action Variables

The variables for this action specify information used to restore the user state of the destination computer, such as pathname of the folder from which the user state is restored and whether the local computer account is restored. For more information about the task sequence step associated with these variables, see Restore User State.

Details

ACTION VARIABLE NAME	DESCRIPTION
OSDStateStorePath (input)	The UNC or local pathname of the folder from which the user state is restored.
OSDMigrateContinueOnRestore (input)	Specifies that the user state restoration continues even if some files cannot be restored. Valid values: **"true"** (default) **"false"**
OSDMigrateEnableVerboseLogging (input)	Enables verbose logging for the USMT tool. This value is required by the action; it must be set to "true" or "false". Valid values: **"true"** **"false"** (default)

ACTION VARIABLE NAME	DESCRIPTION
OSDMigrateLocalAccounts (input)	Specifies whether the local computer account is restored. Valid values: **"true"** **"false"** (default)
OSDMigrateLocalAccountPassword (input)	If the **OSDMigrateLocalAccounts** variable is "true," this variable must contain the password that is assigned to all local accounts that are migrated. Because the same password is assigned to all migrated local accounts, it is considered a temporary password that will be changed later by some method other than Configuration Manager operating system deployment.
OSDMigrateAdditionalRestoreOptions (input)	Specifies additional user state migration tool (USMT) command line options that are used when restoring the user state. The additional options are specified in the form of a string that is appended to the automatically generated USMT command line. The USMT options specified with this task sequence variable are not validated for accuracy prior to running the task sequence.
_OSDMigrateUsmtRestorePackageID (input)	Specifies the package ID of the Configuration Manager package that contains the USMT files. This variable is required.

Run Command Line Task Sequence Action Variables

The variables for this action specify information used to run a command from the command line, such as the working directory where the command is run. For more information about the task sequence step associated with these variables, see Run Command Line.

Details

ACTION VARIABLE NAME	DESCRIPTION
SMSTSDisableWow64Redirection (input)	By default, when running on a 64-bit operating system, the program in the command line is located and run using the WOW64 file system redirector so that 32-bit versions of operating system programs and DLLs are found. Setting this variable to "true" disables the use of the WOW64 file system redirector so that native 64-bit versions of operating system programs and DLLs can be found. This variable has no effect when running on a 32-bit operating system.
WorkingDirectory (input)	Specifies the starting directory for a command-line action. The specified directory name must not exceed 255 characters. Examples: - **"C:\"** - **"%SystemRoot%"**
SMSTSRunCommandLineUserName (input)	Specifies the account by which the command line is run. The value is a string of the form username or domain\username.

ACTION VARIABLE NAME	DESCRIPTION
SMSTSRunCommandLinePassword (input)	Specifies the password for the account specified by the SMSTSRunCommandLineUserName variable.

Set Dynamic Variables

The variables for this action are automatically set when you add the Set Dynamic Variables task sequence step. For more information about the task sequence step associated with these variables, see Set Dynamic Variables.

Details

ACTION VARIABLE NAME (INPUT)	DESCRIPTION
_SMSTSMake	Specifies the make of the computer.
_SMSTSModel	Specifies the model of the computer.
_SMSTSMacAddresses	Specifies the MAC addresses used by the computer.
_SMSTSIPAddresses	Specifies the IP addresses used by the computer.
_SMSTSSerialNumber	Specifies the serial number of the computer.
_SMSTSAssetTag	Specifies the asset tag for the computer.
_SMSTSUUID	Specifies the UUID of the computer.
_SMSTSDefaultGateways	Specifies the default gateways used by the computer.

Setup Windows and ConfigMgr Task Sequence Action Variables

The variable for this action specifies the client installation properties that are used when installing the Configuration Manager client. For more information about the task sequence step associated with these variables, see Setup Windows and ConfigMgr.

Details

ACTION VARIABLE NAME (INPUT)	DESCRIPTION
SMSClientInstallProperties (input)	Specifies the client installation properties that are used when installing the Configuration Manager client.

Upgrade Operating System

The variable for this action specifies additional command-line options that are not available in the Configuration Manager console are added to Setup for a Windows 10 upgrade. For more information about the task sequence step associated with this variable, see Upgrade Operating System.

Details

ACTION VARIABLE NAME (INPUT)	DESCRIPTION
OSDSetupAdditionalUpgradeOptions (input)	Specifies the additional command-line options that are added to Setup during a Windows 10 upgrade. The command-line options are not verified. Therefore, check that the option you enter is accurate. For more information, see Windows Setup Command-Line Options.

Task sequence built-in variables in System Center Configuration Manager

3/26/2017 • 14 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Task sequence built-in variables are provided by System Center Configuration Manager. Built-in variables provide information about the environment where the task sequence is running, and their values are available throughout the whole task sequence. Typically, built-in variables are initialized before steps are run in the task sequence. For example, the built-in variable **_SMSTSLogPath** is an environment variable that specifies the path that Configuration Manager components use to write log files while the task sequence runs; any task sequence step can access this environment variable. However, some variables, such as _SMSTSCurrentActionName, are evaluated before each step. The values of built-in variables are generally read-only. The values are read only for built-in variables with a name that begins with an underscore.

Task Sequence Built-in Variable List

The following list describes the built-in variables that are available in Configuration Manager:

BUILT-IN VARIABLE NAME	DESCRIPTION
_OSDDetectedWinDir	Beginning in Configuration Manager version 1602, the task sequence scans the computer's hard drives for a previous operating system installation when Windows PE starts. The Windows folder location is stored in this variable. You can configure your task sequence to retrieve this value from the environment and use it to specify the same Windows folder location to use for the new operating system installation.
_OSDDetectedWinDrive	Beginning in Configuration Manager version 1602, the task sequence scans the computer's hard drives for a previous operating system installation when Windows PE starts. The hard drive location for where the operating system is installed is stored in this variable. You can configure your task sequence to retrieve this value from the environment and use it to specify the same hard drive location to use for the new operating system.
_SMSTSAdvertID	Stores the current running task sequence deployment unique ID. It uses the same format as a Configuration Manager software distribution deployment ID. If the task sequence is running from stand-alone media, this variable is undefined. Example: **ABC20001**

BUILT-IN VARIABLE NAME	DESCRIPTION
_TSAppInstallStatus	The task sequence sets the _TSAppInstallStatus variable with the installation status for the application during the Install Application task sequence step. The task sequence sets the variable with one of the following values: 1. **Undefined**: Set when the Install Application task sequence step has not been run. 2. **Error**: Set when at least one application failed because of an error during the Install Application task sequence step. 3. **Warning**: Set when no errors occur during the Install Application task sequence step, but one or more applications, or a required dependency, did not install because a requirement was not met. 4. **Success**: Set when there are no errors or warning detected during the Install Application task sequence step.
_SMSTSBootImageID	Stores the Configuration Manager boot image package ID if a boot image package is associated with the current running task sequence. The variable will not be set if no Configuration Manager boot image package is associated. Example: **ABC00001**
_SMSTSBootUEFI	The task sequence sets the SMSTSBootUEFI variable when it detects a computer that is in UEFI mode.
_SMSTSClientGUID	Stores the value of Configuration Manager client GUID. This variable is not set if the task sequence is running from stand-alone media. Example: **0a1a9a4b-fc56-44f6-b7cd-c3f8ee37c04c**
_SMSTSCurrentActionName	Specifies the name of the currently running task sequence step. This variable is set before the task sequence manager runs each individual step. Example: **run command line**
_SMSTSDownloadOnDemand	Set to **true** if the current task sequence is running in download-on-demand mode, which means the task sequence manager downloads content locally only when it must access the content.
_SMSTSInWinPE	This variable is set to **true** when the current task sequence step is running in the Windows PE environment, and it is set to **false** if not. You can test this task sequence variable to determine the current operating system environment.

BUILT-IN VARIABLE NAME	DESCRIPTION
_SMSTSLastActionRetCode	Stores the return code that was returned by the last action that was run. This variable can be used as a condition to determine if the next step is run. Example: **0**
_SMSTSLastActionSucceeded	The variable is set to **true** if the last action succeeded and to **false** if the last action failed. If the last action was skipped because the step was disabled or the associated condition evaluated to **false**, this variable is not reset, which means it still holds the value for the previous action.
_SMSTSLaunchMode	Specifies the task sequence launch method. The task sequence can have the following values: - **SMS** - specifies that the task sequence is started by using the Configuration Manager client. - **UFD** - specifies that the task sequence is started by using USB media and that the USB media was created in Windows XP/2003. - **UFD+FORMAT** - specifies that the task sequence is started by using USB media and that the USB media was created in Windows Vista or later. - **CD** - specifies that the task sequence is started by using a CD. - **DVD** - specifies that the task sequence is started by using a DVD. - **PXE** - specifies that the task sequence is started from PXE. - **HD** - specifies that the task sequence was started from a hard disk (prestaged media only).
_SMSTSLogPath	Stores the full path of the log directory. This can be used to determine where actions are logged. This value is not set when a hard drive is not available.
_SMSTSMachineName	Stores and specifies the computer name. Stores the name of the computer that the task sequence will use to log all status messages. To change the computer name in the new operating system, use the **OSDComputerName** variable. Example: **ABC**
_SMSTSMDataPath	Specifies the path defined by the SMSTSLocalDataDrive variable. When you define SMSTSLocalDataDrive before the task sequence starts, such as by setting a collection variable, Configuration Manager then defines the _SMSTSMDataPath variable once the Task Sequence starts.
_SMSTSMediaType	Specifies the type of media that is used to initiate the installation. Examples of types of media are Boot Media, Full Media, PXE, and Prestaged Media.
_SMSTSMP	Stores the URL or IP address of a Configuration Manager management point.

BUILT-IN VARIABLE NAME	DESCRIPTION
_SMSTSMPPort	Stores the management point port number of a Configuration Manager management point. Example: **80**
_SMSTSOrgName	Stores the branding title name that is displayed in a task sequence progress user interface dialog box. Example: **XYZ Organization**
_SMSTSOSUpgradeActionReturnCode	Stores the exit code value returned from Setup to indicate success or failure. This variable is set during the task sequence steps Operating System Upgrade task sequence step. This is useful with the /Compat Windows 10 Setup command-line option. Example: On the completion of /Compat, you can take actions in later steps depending on the failure or success exit code. On success, you could initiate the upgrade. Or, you could set a marker in the environment (for example, add a file or set a registry key) that can then be used to create a collection of computers that are ready to upgrade or that require action before they are upgraded.
_SMSTSPackageID	Stores the current running task sequence ID. This ID uses the same format as a Configuration Manager software package ID. Example: **HJT00001**
_SMSTSPackageName	Stores the current running task sequence name specified by the Configuration Manager administrator when the task sequence is created. Example: **Deploy Windows 10 task sequence**
_SMSTSSetupRollback	Specifies whether the operating system Setup performed a rollback operation. The variable values can be **true** or **false**.
_SMSTSRunFromDP	Set to **true** if the current task sequence is running in run-from-distribution-point mode, which means the task sequence manager obtains required package shares from distribution point.

BUILT-IN VARIABLE NAME	DESCRIPTION
_SMSTSSiteCode	Stores the site code of the Configuration Manager site. Example: **ABC**
_SMSTSType	Specifies the type of the current running task sequence. It can have the following values: **1** - indicates a generic task sequence. **2** - indicates an operating system deployment task sequence.
_SMSTSTimezone	The _SMSTSTimezone variable stores the time zone information in the following format (without spaces): Bias, StandardBias, DaylightBias, StandardDate.wYear, wMonth, wDayOfWeek, wDay, wHour, wMinute, wSecond, wMilliseconds, DaylightDate.wYear, wMonth, wDayOfWeek, wDay, wHour, wMinute, wSecond, wMilliseconds, StandardName, DaylightName Example: For the Eastern Time U.S. and Canada, the value would be **300,0,-60,0,11,0,1,2,0,0,0,0,3,0,2,2,0,0,0,Eastern Standard Time,Eastern Daylight Time**
_SMSTSUseCRL	Specifies whether the task sequence uses the certificate revocation list when it uses a Secure Socket Layer (SSL) certificate to communicate with the management point.
_SMSTSUserStarted	Specifies whether a task sequence is started by a user. This variable is set only if the task sequence is started from the Software Center. For example, if **_SMSTSLaunchMode** is set to **SMS**. The variable can have the following values: - **true** - specifies that the task sequence is manually started by a user from the Software Center. - **false** - specifies that the task sequence is initiated automatically by the Configuration Manager scheduler.
_SMSTSUseSSL	Specifies whether the task sequence uses SSL to communicate with the Configuration Manager management point. If your site is running in native mode, the value is set to **true**.
_SMSTSWTG	Specifies if the computer is running as a Windows To Go device.

BUILT-IN VARIABLE NAME	DESCRIPTION
OSDPreserveDriveLetter	Beginning in Configuration Manager version 1606, this task sequence variable has been deprecated. During an operating system deployment, by default, Windows Setup determines the best drive letter to use (typically C:). In previous versions, the OSDPreverveDriveLetter variable determines whether or not the task sequence uses the drive letter captured in the operating system image WIM file when applying that image to a destination computer. You can set the value for this variable to **False** to use the location that you specify for the **Destination** setting in the **Apply Operating System** task sequence step. For more information, see Apply Operating System Image.
OSDSetupAdditionalUpgradeOptions	Beginning in Configuration Manager version 1602, you can use this variable to specify additional options for Windows Setup upgrade.
SMSTSAssignmentsDownloadInterval	Use this variable to specify the number of seconds to wait before the client will attempt to download the policy since the last attempt (which returned no policies). By default, the client will wait **0** seconds before retrying. You can set this variable by using a prestart command from media or PXE.
SMSTSAssignmentsDownloadRetry	Use this variable to specify the number of times a client will attempt to download the policy after no policies are found on the first attempt. By default, the client will retry **0** times. You can set this variable by using a prestart command from media or PXE.
SMSTSAssignUsersMode	Specifies how a task sequence associates users with the destination computer. Set the variable to one of the following values. - Auto: The task sequence creates a relationship between the specified users and destination computer when it deploys the operating system to the destination computer. - Pending: The task sequence creates a relationship between the specified users and the destination computer, but waits for approval from the administrative user before the relationship is set. - Disabled: The task sequence does not associate users with the destination computer when it deploys the operating system.
SMSTSDownloadAbortCode	This variable contains the abort code value for the external program downloader (specified in the SMSTSDownloadProgram variable). If the program returns an error code equal to the value of the SMSTSDownloadAbortCode variable, then the content download fails and no other download method is attempted.

BUILT-IN VARIABLE NAME	DESCRIPTION
SMSTSDownloadProgram	Use this variable to specify an Alternate Content Provider, a downloader program that is used to download content instead of the default Configuration Manager downloader, for the task sequence. As part of the content download process, the task sequence checks the variable for a specified downloader program. If specified, the task sequence runs the program to perform the download.
SMSTSDownloadRetryCount	Use this variable to specify the number of times that Configuration Manager attempts to download content from a distribution point. By default, the client will retry **2** times.
SMSTSDownloadRetryDelay	Use this variable to specify the number of seconds that Configuration Manager waits before it retries to download content from a distribution point. By default, the client will wait **15** seconds before retrying.
SMSTSErrorDialogTimeout	When an error occurs in a task sequence, a dialog box is displayed that is automatically dismissed after a number of seconds specified by this variable. By default, the dialog box is automatically dismissed after **900** seconds (15 minutes).
TSErrorOnWarning	Use this variable to specify whether the task sequence engine considers a detected warning as an error during the Application Installation task sequence step. The task sequence sets the _TSAppInstallStatus variable to **Warning** when one or more applications, or a required dependency, did not install because a requirement was not met. When you set the TSErrorOnWarning variable to **True** and the _TSAppInstallStatus variable is set to Warning, it is treated as an error. A value of **False** is the default behavior.
SMSTSLanguageFolder	Use this variable to change the display language of a language neutral boot image.
SMSTSLocalDataDrive	Specifies where temporary files are stored on the destination computer while the task sequence is running. This variable must be set before the task sequence starts, such as by setting a collection variable. Once the task sequence starts, Configuration Manager defines the _SMSTSMDataPath variable once the Task Sequence starts.
SMSTSMP	Use this variable to specify the URL or IP address of the Configuration Manager management point.
SMSTSMPListRequestTimeout	Use this variable to specify how many milliseconds a task sequence waits before it retries to install an application or software update after it fails to retrieve the management point list from location services. By default, the task sequence waits 60,000 milliseconds (60 seconds) before it retries the step, and retries up to three times. This variable is applicable only to the Install Application and Install Software Updates task sequence steps.

BUILT-IN VARIABLE NAME	DESCRIPTION
SMSTSMPListRequestTimeoutEnabled	Use this variable to enable repeated MPList requests to refresh the client if the client is not on the Intranet. By default, this variable is set to True. When clients are on the internet, you can set this variable to False to avoid unnecessary delays. This variable is applicable only to the Install Application and Install Software Updates task sequence steps.
SMSTSPeerDownload	Use this variable to enable the client to use Windows PE Peer Cache. Example: SMSTSPeerDownload = **TRUE** enables this functionality.
SMSTSPeerRequestPort	Use this variable for Windows PE peer cache to specify a custom network port to use for the initial broadcast when you do not use the default ports configured in the Client Settings (8004).
SMSTSPersistContent	Use this variable to temporarily persist content in the task sequence cache.
SMSTSPostAction	Specifies a command that is run after the task sequence completes. For example, you can use this variable to specify a script that enables write filters on embedded devices after the task sequence deploys an operating system to the device.
SMSTSPreferredAdvertID	Forces a specific targeted deployment on the destination computer to be run. This can be set through a prestart command from media or PXE. If this variable is set, the task sequence overrides any required deployments.
SMSTSPreserveContent	This variable flags the content in the task sequence to be retained in the Configuration Manager client cache after the deployment. This is different than using SMSTSPersistContent that only preserves the content for the duration of the task sequence and uses the task sequence cache, not the Configuration Manager client cache. Example: SMSTSPreserveContent = **TRUE** enables this functionality.
SMSTSRebootDelay	Specifies how many seconds to wait before the computer restarts. The task sequence manager will display a notification dialog before reboot if this variable is not set to 0. Examples: **0** **30**

BUILT-IN VARIABLE NAME	DESCRIPTION
SMSTSRebootMessage	Specifies the message to display in the shutdown dialog box when a restart is requested. If this variable is not set, a default message will appear. Example: **This computer is being restarted by the task sequence manager**.
SMSTSRebootRequested	Indicates that a restart is requested after the current task sequence step is completed. If a restart is required, just set this variable to **true**, and the task sequence manager will restart the computer after this task sequence step. The task sequence step must set this task sequence variable if it requires the restart to complete the task sequence step. After the computer is restarted, the task sequence will continue to run from the next task sequence step.
SMSTSRetryRequested	Requests a retry after the current task sequence step is completed. If this task sequence variable is set, the **SMSTSRebootRequested** must also be set to **true**. After the computer is restarted, the task sequence manager will rerun the same task sequence step.
SMSTSSoftwareUpdateScanTimeout	Gives you the ability to control the timeout for the software updates scan during the Install Software Updates task sequence step. For example, you might increase the default value if you have a lot of software updates to install. The default value is 30 minutes.
SMSTSUDAUsers	Specifies the primary user of the destination computer. Specify the users by using the following format. Separate multiple users by using a comma (,). Example: **domain\user1, domain\user2, domain\user3** For more information about associating users with the destination computer, see Associate users with a destination computer.
SMSTSWaitForSecondReboot	Beginning in Configuration Manager version 1602, this optional task sequence variable is available to help control client behavior when a software update installation requires two restarts. This variable must be set before the Install Software Updates step to prevent a task sequence from failing because of a second restart from software update installation. Set the SMSTSWaitForSecondReboot value in seconds to specify how long the task sequence pauses during the Install Software Updates step when the computer restarts to allow sufficient time in case there is a second restart. For example, if you set SMSTSWaitForSecondReboot to 600, the task sequence is paused for 10 minutes after a restart before additional task sequence steps run. This is useful when hundreds of software updates are installed in a single Install Software Updates task sequence step.

Prestart commands for task sequence media in System Center Configuration Manager

11/23/2016 • 2 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can create a prestart command in System Center Configuration Manager to use with boot media, stand-alone media, and prestaged media. The prestart command is a script or executable that runs before the task sequence is selected and can interact with the user in Windows PE. The prestart command can prompt a user for information and save it in the task sequence environment or query a task sequence variable for information. When the destination computer boots, the command-line is run before the policy is downloaded from the management point. Use the following procedures to create a script to use for the prestart command, distribute the content associated with the prestart command, and configure the prestart command in media.

Create a script file to use for the Prestart Command

Task sequence variables can be read and written by using the Microsoft.SMS.TSEnvironment COM object while the task sequence is running. The following example illustrates a Visual Basic script file that queries the _SMSTSLogPath task sequence variable to get the current log location. The script also sets a custom variable.

```
dim osd: set env = CreateObject("Microsoft.SMS.TSEnvironment")
dim logPath
' You can query the environment to get an existing variable.
logPath = env("_SMSTSLogPath")
' You can also set a variable in the OSD environment.
env("MyCustomVariable") = "varname"
```

Create a Package for the Script File and Distribute the Content

After you create the script or executable for the prestart command, you must create a package source to host the files for the script or executable, create a package for the files (no program required), and then distribute the content to a distribution point.

For more information about creating a package, see Packages and programs.

For more information about distributing content, see Distribute content.

Configure the Prestart Command in Media

You can configure a prestart command in the Create Task Sequence Media Wizard for stand-alone media, bootable media, or prestaged media. For more information about the media types, see Create task sequence media. Use the following procedure to create a prestart command in media.

To create a prestart command in media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, select **Stand-alone media**, **Bootable media**, or **Prestaged media**, and

then click **Next**.

5. Navigate to the **Customization** page of the wizard. For more information about configuring the other pages in the wizard, see Create task sequence media.

6. On the **Customization** page, specify the following information, and then click **Next**.

   - Select **Enable prestart command**.
   - In the **Command line** text box, enter the script or executable that you created for the prestart command.

     > **IMPORTANT**
     > Use **cmd /C** to specify the prestart command. For example, if you used TSScript.vbs as the name for your prestart command script, you would enter **cmd /C TSScript.vbs** for the command line. Where **cmd /C** opens a new Windows command interpreter window and uses the Path environment variable to find the prestart command script or executable. You can also specify the full path to the prestart command, but the drive letter could be different on computers with different drive configurations.

   - Select **Include files for the prestart command**.
   - Click **Set** to select the package that is associated with the prestart command files.
   - Click **Browse** to select the distribution point that hosts the content for the prestart command.

7. Complete the wizard.

Infrastructure requirements for operating system deployment in System Center Configuration Manager

3/21/2017 • 10 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Operating system deployment in System Center 2012 Configuration Manager has external dependencies and dependencies within the product. Use the following sections to help you prepare for operating system deployment.

Dependencies External to Configuration Manager

The following provides information about external tools, installation kits, and operating systems that are required to deploy operating systems in Configuration Manager.

Windows ADK for Windows 10

Windows ADK is a set of tools and documentation that support the configuration and deployment of Windows operating systems. Configuration Manager uses Windows ADK to automate Windows installations, capture Windows images, migrate user profiles and data, and so on.

The following features of the Windows ADK must be installed on site server of the top-level site of the hierarchy, on the site server of each primary site in the hierarchy, and on the SMS Provider site system server:

- User State Migration Tool (USMT) [1]
- Windows Deployment Tools
- Windows Preinstallation Environment (Windows PE)

[1] USMT is not required on the SMS Provider site system server.

> **NOTE**
> You must manually install the Windows ADK on each computer that will host a central administration site or primary site server before you install the Configuration Manager site.

For more information, see:

- Windows ADK for Windows 10 scenarios for IT Pros
- Download the Windows ADK for Windows 10

User State Migration Tool (USMT)

Configuration Manager uses a USMT package that contains the USMT 10 source files to capture and restore the user state as part of your operating system deployment. Configuration Manager Setup at the top-level site automatically creates the USMT package. USMT 10 can capture user state from Windows 7, Windows 8, Windows 8.1, and Windows 10. USMT 10 is distributed in the Windows Assessment and Deployment Kit (Windows ADK) for Windows 10.

For more information, see:

- Common Migration Scenarios for USMT 10

- Manage user state

Windows PE

Windows PE is used for boot images to start a computer. It is a Windows operating system with limited services that is used during the pre-installation and deployment of Windows operating systems. The following provides the version of Configuration Manager and the supported version of Windows ADK, the Windows PE version on which the boot image is based that can be customized from the Configuration Manager console, and the Windows PE versions on which the boot image is based that you can customize by using DISM and then add the image to the specified version of Configuration Manager.

Configuration Manager version 1511

The following provides the supported version of Windows ADK, the Windows PE version on which the boot image is based that can be customized from the Configuration Manager console, and the Windows PE versions on which the boot image is based that you can customize by using DISM and then add the image to Configuration Manager.

- **Windows ADK version**

  Windows ADK for Windows 10

- **Windows PE versions for boot images customizable from the Configuration Manager console**

  Windows PE 10

- **Supported Windows PE versions for boot images not customizable from the Configuration Manager console**

  Windows PE 3.1[1] and Windows PE 5

  [1] You can only add a boot image to Configuration Manager when it is based on Windows PE 3.1. Install the Windows AIK Supplement for Windows 7 SP1 to upgrade Windows AIK for Windows 7 (based on Windows PE 3) with the Windows AIK Supplement for Windows 7 SP1 (based on Windows PE 3.1). You can download Windows AIK Supplement for Windows 7 SP1 from the Microsoft Download Center.

  For example, when you have Configuration Manager, you can customize boot images from Windows ADK for Windows 10 (based on Windows PE 10) from the Configuration Manager console. However, while boot images based on Windows PE 5 are supported, you must customize them from a different computer and use the version of DISM that is installed with Windows ADK for Windows 8. Then, you can add the boot image to the Configuration Manager console. For more information with the steps to customize a boot image (add optional components and drivers), enable command support to the boot image, add the boot image to the Configuration Manager console, and update distribution points with the boot image, see Customize boot images. For more information about boot images, see Manage boot images.

Windows Server Update Services (WSUS)

You must install the following WSUS 4.0 hotfixes:

- Hotfix 3095113 is necessary for Windows 10 servicing, which uses the software updates infrastructure to get Windows 10 feature upgrades. When you have WSUS 3.2, you must use task sequences to upgrade Windows 10. For more information, see Manage Windows as a service.
- Hotfix 3159706 is necessary to use Windows 10 servicing to upgrade computers to the Windows 10 Anniversary Update, as well as for subsequence versions. There are manual steps described in the support article that you must take to install this hotfix. For more information, see Manage Windows as a service.

Internet Information Services (IIS) on the site system servers

IIS is required for the distribution point, state migration point, and management point. For more information about this requirement, see Site and site system prerequisites.

Windows Deployment Services (WDS)

WDS is needed for PXE deployments, when you use multicast to optimize bandwidth in your deployments, and for offline servicing of images. If the provider is installed on a remote server, you must install WDS on the site server and the remote provider. For more information, see Windows Deployment Services in this topic.

Dynamic Host Configuration Protocol (DHCP)

DHCP is required for PXE deployments. You must have a functioning DHCP server with an active host to deploy operating systems by using PXE. For more information about PXE deployments, see Use PXE to deploy Windows over the network.

Supported operating systems and hard disk configurations

For more information about the operating system versions and hard disk configurations that are supported by Configuration Manager when you deploy operating systems, see Supported Operating Systems and Supported Disk Configurations.

Windows device drivers

Windows device drivers can be used when you install the operating system on the destination computer and when you run Windows PE by using a boot image. For more information about device drivers, see Manage drivers.

Configuration Manager Dependencies

The following provides information about Configuration Manager operating system deployment prerequisites.

Operating system image

Operating system images in Configuration Manager are stored in the Windows Imaging (WIM) file format and represent a compressed collection of reference files and folders that are required to successfully install and configure an operating system on a computer. For more information, see Manage operating system images.

Driver catalog

To deploy a device driver, you must import the device driver, enable it, and make it available on a distribution point that the Configuration Manager client can access. For more information about the driver catalog, see Manage drivers.

Management point

Management points transfer information between client computers and the Configuration Manager site. The client uses a management point to run any task sequences that are required to complete the operating system deployment.

For more information about task sequences, see Planning considerations for automating tasks.

Distribution point

Distribution points are used in most deployments to store the data that is used to deploy an operating system, such as the operating system image or device driver packages. Task sequences typically retrieve data from a distribution point to deploy the operating system.

For more information about how to install distribution points and manage content, see Manage content and content infrastructure.

PXE-enabled distribution point

To deploy PXE-initiated deployments, you must configure a distribution point to accept PXE requests from clients. For more information about how to configure the distribution point, see Configure a distribution point.

Multicast-enabled distribution point

To optimize your operating system deployments by using multicast, you must configure a distribution point to support multicast. For more information about how to configure the distribution point, see Configure a distribution point.

State migration point

When you capture and restore user state data for side-by-side and refresh deployments, you must configure a state migration point to store the user state data on another computer.

For more about how to configure the state migration point, see State migration point.

For information about how to capture and restore user state, see Manage user state.

Service connection point

When you use Windows as a Service (WaaS) to deploy Windows 10 Current Branch, you must have the service connection point installed. For more information, see Manage Windows as a service.

Reporting services point

To use Configuration Manager reports for operating system deployments, you must install and configure a reporting services point. For more information, see Reporting.

Security permissions for operating system deployments

The **Operating System Deployment Manager** security role is a built-in role that cannot be changed. However, you can copy the role, make changes, and then save these changes as a new custom security role. Here are some of the permissions that apply directly to operating system deployments:

- **Boot Image Package**: Create, Delete, Modify, Modify Folder, Move Object, Read, Set Security Scope
- **Device Drivers**: Create, Delete, Modify, Modify Folder, Modify Report, Move Object, Read, Run Report
- **Driver Package**: Create, Delete, Modify, Modify Folder, Move Object, Read, Set Security Scope
- **Operating System Image**: Create, Delete, Modify, Modify Folder, Move Object, Read, Set Security Scope
- **Operating System Installation Package**: Create, Delete, Modify, Modify Folder, Move Object, Read, Set Security Scope
- **Task Sequence Package**: Create, Create Task Sequence Media, Delete, Modify, Modify Folder, Modify Report, Move Object, Read, Run Report, Set Security Scope

For more information about custom security roles, see Create custom security roles.

Security scopes for operating system deployments

Use security scopes to provide administrative users with access to the securable objects used in operating system deployments, such as operating system and boot images, driver packages, and task sequence packages. For more information, see Security scopes.

Windows Deployment Services

Windows Deployment Services (WDS) must be installed on the same server as the distribution points that you configure to support PXE or multicast. WDS is included in the operating system of the server. For PXE deployments, WDS is the service that performs the PXE boot. When the distribution point is installed and enabled for PXE, Configuration Manager installs a provider into WDS that uses the WDS PXE boot functions.

> **NOTE**
> The installation of WDS might fail if the server requires a restart.

Other WDS configurations that must be considered include the following:

- The WDS installation on the server requires that the administrator is a member of the Local Administrators group.

- The WDS server must be either a member of an Active Directory domain or a domain controller for an Active Directory domain. All Windows domain and forest configurations support WDS.
- If the provider is installed on a remote server, you must install WDS on the site server and the remote provider.

Considerations when you have WDS and DHCP on the same server

Consider the following configuration issues if you plan to co-host the distribution point on a server running DHCP.

- You must have a functioning DHCP server with an active scope. Windows Deployment Services uses PXE, which requires a DHCP server.
- DHCP and Windows Deployment Services both require port number 67. If you co-host Windows Deployment Services and DHCP, you can move DHCP or the distribution point that is configured for PXE to a separate server. Or, you can use the following procedure to configure the Windows Deployment Services server to listen on a different port.

  To configure the Windows Deployment Services server to listen on a different port

  1. Modify the following registry key:

     HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WDSServer\Providers\WDSPXE

  2. Set the registry value to: **UseDHCPPorts = 0**
  3. For the new configuration to take effect, run the following command on the server:

     ```
     WDSUTIL /Set-Server /UseDHCPPorts:No /DHCPOption60:Yes
     ```

- A DNS server is required to run Windows Deployment Services.
- The following UDP ports must be open on the Windows Deployment Services server.
  - Port 67 (DHCP)
  - Port 69 (TFTP)
  - Port 4011 (PXE)

  > **NOTE**
  > In addition, if DHCP authorization is required on the server, you need DHCP client port 68 to be open on the server.

Supported Operating Systems

All Windows operating systems listed as supported client operating systems in Supported operating systems for clients and devices are supported for operating system deployments.

Supported Disk Configurations

The hard disk configuration combinations on the reference and destination computers that are supported for Configuration Manager operating system deployment are shown in the following table.

REFERENCE COMPUTER HARD DISK CONFIGURATION	DESTINATION COMPUTER HARD DISK CONFIGURATION
Basic disk	Basic disk
Simple volume on a dynamic disk	Simple volume on a dynamic disk

Configuration Manager supports capturing an operating system image only from computers that are configured with simple volumes. There is no support for the following hard disk configurations:

- Spanned volumes
- Striped volumes (RAID 0)
- Mirrored volumes (RAID 1)
- Parity volumes (RAID 5)

The following table shows an additional hard disk configuration on the reference and destination computers that is not supported with Configuration Manager operating system deployment.

REFERENCE COMPUTER HARD DISK CONFIGURATION	DESTINATION COMPUTER HARD DISK CONFIGURATION
Basic disk	Dynamic disk

Next steps

Prepare for operating system deployment

Planning considerations for automating tasks in System Center Configuration Manager

11/23/2016 • 26 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can create task sequences to automate tasks in your System Center Configuration Manager environment. These tasks range from capturing an operating system on a reference computer to deploying the operating system to one or more destination computers. The actions of the task sequence are defined in the individual steps of the sequence. When the task sequence is run, the actions of each step are performed at the command-line level in the Local System context without requiring user intervention. Use the following sections to help plan to automate tasks in Configuration Manager.

Task sequence steps and actions

Steps are the basic components of a task sequence. They can contain commands that configure and capture the operating system of a reference computer, or they can contain commands that install the operating system, drivers, the Configuration Manager client, and software on the destination computer. The commands of a task sequence step are defined by the actions of the step. There are two types of actions. An action that you define by using a command-line string is referred to as a custom action. An action that is predefined by Configuration Manager is referred to as a built-in action. A task sequence can perform any combination of custom and built-in actions.

Task sequence steps can also include conditions that control how the step behaves, such as stopping the task sequence or continuing the task sequence if an error occurs. Conditions are added to the step by including a task sequence variable to the step. For example, you could use the **SMSTSLastActionRetCode** variable to test the condition of the previous step. Variables can be added to a single step or a group of steps.

Task sequence steps are processed sequentially, which includes the action of the step and any conditions that are assigned to the step. When Configuration Manager starts to process a task sequence step, the next step is not started until the previous action has completed. A task sequence is considered complete when all its steps have been completed or when a failed step causes Configuration Manager to stop running the task sequence before all its steps are completed. For example, if the step of a task sequence cannot locate a referenced image or package on a distribution point, the task sequence contains a broken reference and Configuration Manager stops running the task sequence at that point unless the failed step has a condition to continue when an error occurs.

> **IMPORTANT**
> By default, a task sequence fails after one step or action fails. If you want the task sequence to continue even when a step fails, edit the task sequence, click the **Options** tab, and then select **Continue on error**.

For more information about the steps that can be added to a task sequence, see Task sequence steps.

Task sequence groups

Groups are multiple steps within a task sequence. A task sequence group consists of a name, an optional description, and any optional conditions that are evaluated as a unit before that task sequence continues with the next step. Groups can be nested within each other, and a group can contain a mixture of steps and subgroups. Groups are useful for combining multiple steps that share a common condition.

IMPORTANT

By default, a task sequence group fails when any step or embedded group within the group fails. If you want the task sequence to continue when a step or embedded group fails, edit the task sequence, click the **Options** tab, and then select **Continue on error**.

The following table shows how the **Continue on error** option works when you group steps.

In this example, there are two groups of task sequences that contain three task sequence steps each.

TASK SEQUENCE GROUP OR STEP	CONTINUE ON ERROR SETTING
Task Sequence Group 1	**Continue on error** selected.
Task Sequence Step 1	**Continue on error** selected.
Task Sequence Step 2	Not set.
Task Sequence Step 3	Not set.
Task Sequence Group 2	Not set.
Task Sequence Step 4	Not set.
Task Sequence Step 5	Not set.
Task Sequence Step 6	Not set.

- If task sequence step 1 fails, the task sequence continues with task sequence step 2.
- If task sequence step 2 fails, the task sequence does not run task sequence step 3 but continues to run task sequence steps 4 and 5, which are in a different task sequence group.
- If task sequence step 4 fails, no more steps are run, and the task sequence fails because the **Continue on error** setting was not configured for task sequence group 2.

You must assign a name to task sequence groups, although the group name does not have to be unique. You can also provide an optional description for the task sequence group.

Task sequence variables

Task sequence variables are a set of name and value pairs that supply configuration and operating system deployment settings for computer, operating system, and user state configuration tasks on a Configuration Manager client computer. Task sequence variables provide a mechanism to configure and customize the steps in a task sequence.

When you run a task sequence, many of the task sequence settings are stored as environment variables. You can access or change the values of built-in task sequence variables, and you can create new task sequence variables to customize the way a task sequence runs on a destination computer.

You can use task sequence variables in the task sequence environment to perform the following actions:

- Configure settings for a task sequence action
- Supply command-line arguments for a task sequence step

- Evaluate a condition that determines whether a task sequence step or group is run
- Provide values for custom scripts used in a task sequence

For example, you might have a task sequence that includes a **Join Domain or Workgroup** task sequence step. The task sequence might be deployed to different collections, where the membership of the collection is determined by domain membership. In that case, you can specify a per-collection task sequence variable for each collection's domain name and then use that task sequence variable to supply the appropriate domain name in the task sequence.

Create task sequence variables

You can add new task sequence variables to customize and control the steps in a task sequence. For example, you can create a task sequence variable to override a setting for a built-in task sequence step. You can also create a custom task sequence variable to use with conditions, command lines, or custom steps in the task sequence. When you create a task sequence variable, the task sequence variable and the associated value is preserved within the task sequence environment, even when the sequence restarts the destination computer. The variable and its value can be used within the task sequence across different operating system environments. For example, it can be used in a full Windows operating system and in the Windows PE environment.

The following table describes the methods to create a task sequence variable and additional usage information.

CREATE METHOD	USAGE
Setting fields in task sequence steps by using the Task Sequence Editor	Specifies default values for the task sequence step. The variable and value are accessible only when the step runs in the task sequence. They are not part of the overall sequence environment, and they are not accessible by other task sequence steps in the task sequence. For a list of the built-in variables and their associated actions, see Task sequence action variables.
Adding a set task sequence variable step in a task sequence	Specifies the task sequence variable and value in the task sequence environment when the task sequence step is run as part of a task sequence. All subsequent task sequence steps can access the environment variable and its value.
Defining a per-collection variable	Specifies task sequence variables and values for a collection of computers. All task sequences targeted to the collection can access the task sequence variables and their values.
Defining a per-computer variable	Specifies task sequence variables and values for a particular computer. All task sequences targeted to the computer can access the task sequence variables and their values.
Adding a task sequence variable on the **Customization** page of the Task Sequence Media Wizard	Specifies task sequence variables and values for the task sequence that is run from the media that can access the task sequence variable and its value.

To override the default value for a built-in task sequence variable, you must define a task sequence variable with the same name as the built-in task sequence variable. For a list of built-in task sequence variables with the associated actions and usage, see Task sequence built-in variables.

You can delete a task sequence variable from the task sequence environment by using the same methods as creating a task sequence variable. In this case, to delete a variable from the task sequence environment, you set the task sequence variable value to an empty string.

You can combine methods to set an environment task sequence variable to different values for the same sequence.

In an advanced scenario, you might set the default values for steps in a sequence using the Task Sequence Editor and then set a custom variable value using the different creation methods. The following list describes the rules that determine which value is used when a task sequence variable is created by using more than one method.

1. The **Set Task Sequence Variable** step overrides all other creation methods.
2. Per-computer variables take precedence over per-collection variables. If you specify the same task sequence variable name for a per-computer variable and a per-collection variable, the per-computer variable value is used when the destination computer runs the deployed task sequence.
3. Task sequences can be run from media. Use the media variables in place of per-collection or per-computer variables. If the task sequence is running from media, per-computer and per-collection variables do not apply and are not used. Instead, task sequence variables defined on the **Customization** page of the Task Sequence Media wizard are used to set values specific to a task sequence that runs from media
4. If a task sequence variable value is not set in the overall sequence environment, built-in actions use the default value for the step, as set in the Task Sequence Editor.

In addition to overriding values for built-in task sequence step settings, you can also create a new environment variable for use in a task sequence step, script, command line, or condition. When you specify a name for a new task sequence variable, follow these guidelines:

- The task sequence variable name that you specify can contain letters, numbers, the underscore character (_), and a hyphen (-).
- Task sequence variable names have a minimum length of 1 character and a maximum length of 256 characters.
- User defined variables must begin with a letter (A-Z or a-z).
- User-defined variable names cannot begin with the underscore character. Only read-only task sequence variables are preceded by the underscore character

  > **NOTE**
  > Read-only task sequence variables can be read by task sequence steps in a task sequence but they cannot be set. For example, you can use a read-only task sequence variable as part of the command line for a **Run Command Line** task sequence action variable, but you cannot set a read-only variable by using the **Set Task Sequence Variable** action variable.

- Task sequence variable names are not case sensitive. For example, OSDVAR and osdvar represent the same task sequence variable.
- Task sequence variable names cannot begin or end with a space or contain embedded spaces. Spaces that are left at the beginning or the end of a task sequence variable name are ignored.

The following table displays examples of valid and non-valid user-specified task sequence variables.

EXAMPLES OF VALID USER-SPECIFIED VARIABLE NNAMES	EXAMPLES OF NON VALID USER-SPECIFIED VARIABLE NAMES
MyVariable	1Variable User-specified task sequence variables cannot begin with a number.

EXAMPLES OF VALID USER-SPECIFIED VARIABLE NNAMES	EXAMPLES OF NON VALID USER-SPECIFIED VARIABLE NAMES
My_Variable	MyV@riable User-specified task sequence variables cannot contain the @ symbol.
My_Variable_2	_MyVariable User-specified task sequence variables cannot begin with an underscore.

General limitations for task sequence variables:

- Task sequence variable values cannot exceed 4,000 characters.
- You cannot create or override a read-only task sequence variable. Read-only variables are designated by names that start with an underscore character (_). You can access the value of read-only task sequence variables in your task sequence; however, you cannot change their associated values.
- Task sequence variable values can be case sensitive depending on the usage of the value. In most cases, task sequence variable values are not case sensitive. However, some values can be case sensitive such as a variable that contains a password.
- There is no limit to how many task sequence variables can be created. However, the number of variables is limited by the size of the task sequence environment. The total size limit for the task sequence environment is 32 MB.

Access environment variables

After you specify the task sequence variable and its value by using one of the methods from the previous section, you can use the environment variable value in your task sequences. You can access default values for built-in task sequence variables, specify a new value for a built-in variable, or use a custom task sequence variable in a command line or script.

The following table outlines task sequence operations that can be performed by accessing the task sequence environment variables.

TASK SEQUENCE OPERATION	USAGE
Configure action settings	You can specify that a task sequence step setting is provided by a variable value when the sequence runs. To supply a task sequence step setting by using a task sequence environment variable, use the Task Sequence Editor to edit the step and specify the variable name as the field value. The variable name must be enclosed in percent signs (%) to indicate that it is an environment variable.

TASK SEQUENCE OPERATION	USAGE
Supply command-line arguments	You can specify part or all of a custom command line by using an environment variable value. To supply a command-line setting by using an environment variable, use the variable name as part of the **Command Line** field of the **Run Command Line** task sequence step. The variable name must be enclosed in percent signs (%). For example, the following command line uses a built-in environment variable to write the computer name to C:\File.txt. **Cmd /C %_SMSTSMachineName% > C:\File.txt**
Evaluate a step condition	You can use built-in or custom task sequence environment variables as part of a task sequence step or group condition. The environment variable value will be evaluated before the task sequence step or group runs. To add a condition that evaluates a variable value, do the following: 1. Select the step or group that you want to add the condition to. 2. On the **Options** tab for the step or group, select **Task Sequence Variable** from the **Add Condition** drop down. 3. In the **Task Sequence Variable** dialog box, specify the name of the variable, the condition that is tested, and the value of the variable.

TASK SEQUENCE OPERATION	USAGE
Provide information for a custom script	Task Sequence variables can be read and written by using the Microsoft.SMS.TSEnvironment COM object while the task sequence is running. The following example illustrates a Visual Basic script file that queries the **_SMSTSLogPath** task sequence variable to get the current log location. The script also sets a custom variable. **dim osd: set env = CreateObject("Microsoft.SMS.TSEnvironment")** **dim logPath** **' You can query the environment to get an existing variable.** **logPath = env("_SMSTSLogPath")** **' You can also set a variable in the OSD environment.** **env("MyCustomVariable") = "varname"** For more information about how to use task sequence variables in scripts, refer to the SDK documentation

Computer and collection variables

You can configure task sequences to run on multiple computers or collections simultaneously. You can specify unique per-computer or per-collection information, such as specify a unique operating system product key or join all the members of a collection to a specified domain.

You can assign task sequence variables to a single computer or a collection. When the task sequence starts to run on the target computer or collection, the values specified are applied to the target computer or collection.

You can specify task sequence variables for a single computer or a collection. When the task sequence starts to run on the target computer or collection, the variables specified are added to the environment and the values are available to all task sequence steps in the task sequence.

> **WARNING**
> If you use the same variable name for both a per-collection and per-computer variable, the computer variable value takes precedence over the collection variable. Task sequence variables that you assign to collections take precedence over built-in task sequence variables.

For more information about how to create task sequence variables for computers and collections, see Create task sequence variables for computers and collections.

Task sequence media variables

You can specify task sequence variables for task sequences that are run from media. When using media to deploy the operating system you add the task sequence variables and specify their values when you create the media; the variables and their values are stored on the media.

> **NOTE**
> Task sequences are stored on stand-alone media. However, all other types of media, such as prestaged media, retrieve the task sequence from a management point.

You can specify task sequence variables on the **Customization** page of the Task Sequence Media Wizard. For information about how to create media, see Create task sequence media.

> **TIP**
> The task sequence writes the package ID and prestart command-line, including the value for any task sequence variables, to the CreateTSMedia.log log file on the computer that runs the Configuration Manager console. You can review this log file to verify the value for the task sequence variables.

Create a task sequence

You create task sequences by using the Create Task Sequence Wizard. The wizard can create built-in task sequences that perform specific tasks or custom task sequences that can perform many different tasks.

For example, you can create task sequences that build and capture an operating system image of a reference computer, install an existing operating system image on a destination computer, or create a custom task sequence that performs a customized task. You can use custom task sequences to perform specialized operating system deployments.

For more information about how to create task sequences, see Create task sequences.

Edit a task sequence

You edit the task sequence by using the **Task Sequence Editor**. The editor can make the following changes to the task sequence:

- You can add or remove steps from the task sequence.
- You can change the order of the steps of the task sequence.
- You can add or remove groups of steps.
- You can specify whether the task sequence continues when an error occurs.
- You can add conditions to the steps and groups of a task sequence.

> **IMPORTANT**
> If the task sequence has any unassociated references to a package or a program as a result of the edit, you must correct the reference, delete the unreferenced program from the task sequence, or temporarily disable the failed task sequence step until the broken reference is corrected or removed.

For more information about how to edit task sequences, see Edit a task sequence.

Deploy a task sequence

You can deploy a task sequence to destination computers that are in any Configuration Manager collection. This includes the **All Unknown Computers** collection that is used to deploy operating systems to unknown computers. However, you cannot deploy a task sequence to user collections.

> **IMPORTANT**
>
> Do not deploy task sequences that install operating systems to inappropriate collections, such as the **All Systems** collection. Be sure that the collection that you deploy the task sequence to contains only those computers where you want the operating system to be installed. To help prevent unwanted operating system deployment, you can manage deployment settings. For more information, see Settings to manage high-risk deployments.

Each destination computer that receives the task sequence runs the task sequence according to the settings specified in the deployment. The task sequences itself does not contain associated files or programs. Any files that are referenced by a task sequence must already be present on the destination computer or reside on a distribution point that clients can access. In addition, the task sequence installs the packages that are referenced by programs, even if the program or package is already installed on the destination computer.

> **NOTE**
>
> In comparison to packages and programs, if the task sequence installs an application, the application installs only if the requirement rules for the application are met and the application is not already installed, based on the detection method that is specified for the application.

The Configuration Manager client runs a task sequence deployment when it downloads client policy. To initiate this action rather than wait until the next polling cycle, see Initiate Policy Retrieval for a Configuration Manager Client.

When you deploy task sequences to Windows Embedded devices that are write filter enabled, you can specify whether to disable the write filter on the device during the deployment and then restart the device after the deployment. If the write filter is not disabled, the task sequence is deployed to a temporary overlay and it will not be available when the device restarts.

> **NOTE**
>
> When you deploy a task sequence to a Windows Embedded device, ensure that the device is a member of a collection that has a configured maintenance window. This allows you to manage when the write filter is disabled and enabled, and when the device restarts.
>
> If clients download task sequences outside of a maintenance window, the task sequence is downloaded twice. In this scenario clients will download the task sequence, disable the write filters, restart the computer, and then download the task sequence again because the task sequence was downloaded to the temporary overlay which is cleared when the device restarts.

For more information about how to deploy task sequences, see the Deploy a task sequence.

Export and import a task sequences

Configuration Manager lets you export and import task sequences. When you export a task sequence, you can include the objects that are referenced by the task sequence. These include an operating system image, a boot image, a client agent package, a driver package, and applications that have dependencies.

> **NOTE**
>
> The export and import process for task sequences is very similar to the export and import process for applications in Configuration Manager.

For more information about how to export and import task sequences, see Export and import task sequences.

Run a task sequence

By default, task sequences always run by using the Local System account. The task sequence command-line step provides the ability to run the task sequence as a different account. When the task sequence is run, the Configuration Manager client first checks for any referenced packages before it starts the steps of the task sequence. If a referenced package is not validated or is not available on a distribution point, the task sequence returns an error for the associated task sequence step.

If a distributed task sequence is configured to download and run, all dependent packages and applications are downloaded to the Configuration Manager client cache. The required packages and applications are obtained from distribution points, and if the Configuration Manager client cache size is too small or the package or application cannot be found, the task sequence fails and a status message is generated. You can also specify that the client downloads the content only when it is required when you select **Download content locally when needed by running task sequence**, or you can use the **Run program from distribution point** option to specify that the client installs the files directly from the distribution point without downloading them into the cache first. The **Run program from distribution point** option is available only if the referenced packages have the setting **Copy the content in this package to a package share on distribution points** enabled on the **Data Access** tab of the **Package** properties.

If a dependent package or application cannot be located by the client running the task sequence, the client immediately sends an error when the deployment is configured as **Available**. However, if the deployment is configured as **Required**, the Configuration Manager client waits and retries to download the content until the deadline, in case the content is not yet replicated to a distribution point that the client can access.

When a task sequence completes successfully or fails, Configuration Manager records this in the Configuration Manager client history. You cannot cancel or stop a task sequence after it is initiated on a computer.

> **IMPORTANT**
> If a task sequence step requires the client computer to restart, the client must be able to boot to a formatted disk partition. Otherwise, the task sequence fails regardless of any error handling that is specified by the task sequence.

When a dependent object of a task sequence, such as a software distribution package, is updated to a newer version, any task sequence that references the package is automatically updated and it references the newest version, regardless of how many updates have been deployed.

> **NOTE**
> Before a Configuration Manager client runs a task sequence, the client checks all task sequences for possible dependencies and the availability of those dependencies on a distribution point. If the client finds a deleted object that the task sequence depends on, the client generates an error and does not run the task sequence.

Run a program before the task sequence is run

You can select a program that runs before the task sequence is run. To specify a program to run first, open the **Properties** dialog box for the task sequence and select the **Advanced** tab to set the following options:

> **IMPORTANT**
> To run a program before the task sequence is run, all content for the task sequence and program must be available on a package share for the package. You configure the package share on the **Data Access** tab in the properties for the package.

- **Run another program first**: Specify that you want another program to run before the task sequence is run.

  > **IMPORTANT**
  > This setting applies only to task sequences that run in the full operating system. Configuration Manager ignores this setting if the task sequence is started by using PXE or boot media.

- **Package**: Specify the package that contains the program.
- **Program**: Specify the program to run.
- **Always run this program first**: Specify that you want Configuration Manager to run this program every time it runs the task sequence on the same client. By default, after a program is run successfully, the program is not run again if the task sequence is rerun on the same client.

If the selected program fails to run on a client, the task sequence is not run.

Use a maintenance window to specify when a task sequence can run

You can specify when the task sequence can run by defining a maintenance window for the collection that contains your destination computers. Maintenance windows are configured with a start date, a start and finish time, and a recurrence pattern. In addition, when you set the schedule for the maintenance window you can specify that the maintenance window applies only to task sequences. For more information, see How to use maintenance windows.

> **IMPORTANT**
> When you configure a maintenance window to run a task sequence, once the task sequences starts it continues to run even if the maintenance window closes. The task sequence will either complete successfully or fail.

Task sequences and the Network Access Account

Although task sequences run only in the context of the Local System account, you might need to configure the Network Access Account in the following circumstances:

- You must configure the Network Access Account correctly or the task sequence will fail if it tries to access Configuration Manager packages on distribution points to complete its task. For more information about the Network Access account, see Network Access Account.

  > **NOTE**
  > The Network Access Account is never used as the security context for running programs, installing applications, installing updates, or running task sequences; however, the Network Access account is used to access the associated resources on the network.

- When you use a boot image to initiate an operating system deployment, Configuration Manager uses the Windows PE environment, which is not a full operating system. The Windows PE environment uses an automatically generated, random name that is not a member of any domain. If you do not configure the Network Access Account correctly, the computer might not have the necessary permissions to access the required Configuration Manager packages to complete the task sequence.

Create media for task sequences

You can write task sequences and their related files and dependencies to several types of media. This includes writing to removable media such as a DVD or CD set or a USB flash drive for capture, stand-alone, and bootable

media, or writing to a Windows Imaging Format (WIM) file for prestaged media.

You can create the following types of media:

- **Capture media**. Capture media captures an operating system image that is configured and created outside the Configuration Manager infrastructure. Capture media can contain custom programs that can run before a task sequence runs. The custom program can interact with the desktop, prompt the user for input values, or create variables to be used by the task sequence.

  For more information, see Create capture media.

- **Stand-alone media**. Stand-alone media contains the task sequence and all associated objects that are necessary for the task sequence to run. Stand-alone media task sequences can run when Configuration Manager has limited or no connectivity to the network. Stand-alone media can be run in the following ways:

  - If the destination computer is not booted, the Windows PE image that is associated with the task sequence is used from the stand-alone media and the task sequence begins.
  - The stand-alone media can be manually started if a user is logged on to the network and initiates the installation.

  > **IMPORTANT**
  > The steps of a stand-alone media task sequence must be able to run without any retrieving any data from the network; otherwise, the task sequence step that tries to retrieve the data fails. For example, a task sequence step that requires a distribution point to obtain a package fails; however if the necessary package is contained on the stand-alone media, the task sequence step succeeds.

  For more information, see Create stand-alone media.

- **Bootable media**. Bootable media contains the required files to start a destination computer so that it can connect to the Configuration Manager infrastructure to determine which task sequences to run based on its membership to a collection. The task sequence and dependent objects are not contained on the media; instead, they are obtained over the network from the Configuration Manager client. This method is useful for new computers or bare-metal deployments, or when no Configuration Manager client or operating system is on the destination computer.

  For more information, see Create bootable media.

- **Prestaged media**. Prestaged media deploys an operating system image to a destination computer that is not provisioned. The prestaged media is stored as a Windows Imaging Format (WIM) file that can be installed on a bare-metal computer by the manufacturer or at an enterprise staging center that is not connected to the Configuration Manager environment.

  For more information, see Create prestaged media.

When you create media, specify a password for the media to control access to the files that are contained on the media. If you specify a password, a user must be present to enter the password at the target computer when the task sequence is run.

When you run a task sequence by using media, the specified computer chip architecture contained on the media will not be recognized and the task sequence attempts to run even if the architecture specified does not match what is actually installed on the target computer. If the chip architecture contained on the media does not match the chip architecture installed on the target computer, the installation fails.

For more information about how to deploy operating systems by using media, see Create task sequence media.

Security and privacy for operating system deployment in System Center Configuration Manager

11/23/2016 • 12 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic contains security and privacy information for operating system deployment in System Center Configuration Manager.

Security best practices for operating system deployment

Use the following security best practices for when you deploy operating systems with Configuration Manager:

- **Implement access controls to protect bootable media**

  When you create bootable media, always assign a password to help secure the media. However, even with a password, only files that contain sensitive information are encrypted and all files can be overwritten.

  Control physical access to the media to prevent an attacker from using cryptographic attacks to obtain the client authentication certificate.

  To help prevent a client from installing content or client policy that has been tampered with, the content is hashed and must be used with the original policy. If the content hash fails or the check that the content matches the policy, the client will not use the bootable media. Only the content is hashed; the policy is not but it is encrypted and secured when you specify a password, which makes it more difficult for an attacker to successfully modify the policy.

- **Use a secured location when you create media for operating system images**

  If unauthorized users have access to the location, they can tamper with the files that you create and also use all the available disk space so that the media creation fails.

- **Protect certificate files (.pfx) with a strong password and if you store them on the network, secure the network channel when you import them into Configuration Manager**

  When you require a password to import the client authentication certificate that you use for bootable media, this helps to protect the certificate from an attacker.

  Use SMB signing or IPsec between the network location and the site server to prevent an attacker from tampering with the certificate file.

- **If the client certificate is compromised, block the certificate from Configuration Manager and revoke it if it is a PKI certificate**

  To deploy an operating system by using bootable media and PXE boot, you must have a client authentication certificate with a private key. If that certificate is compromised, block the certificate in the **Certificates** node in the **Administration** workspace, **Security** node.

- **When the SMS Provider is on a computer or computers other than the site server, secure the communication channel to protect boot images**

  When boot images are modified and the SMS Provider is running on a server that is not the site server, the boot images are vulnerable to attack. Protect the network channel between these computers by using SMB signing or IPsec.

- **Enable distribution points for PXE client communication only on secure network segments**

  When a client sends a PXE boot request, you have no way to ensure that the request is serviced by a valid PXE-enabled distribution point. This scenario has the following security risks:

  - A rogue distribution point that responds to PXE requests could provide a tampered image to clients.
  - An attacker could launch a man-in-the-middle attack against the TFTP protocol that is used by PXE and send malicious code with the operating system files, or she could create a rogue client to make TFTP requests directly to the distribution point.
  - An attacker could use a malicious client to launch a denial of service attack against the distribution point.

  Use defense in depth to protect the network segments where clients will access distribution points for PXE requests.

  > **WARNING**
  > Because of these security risks, do not enable a distribution point for PXE communication when it is in an untrusted network, such as a perimeter network.

- **Configure PXE-enabled distribution points to respond to PXE requests only on specified network interfaces**

  If you allow the distribution point to respond to PXE requests on all network interfaces, this configuration might expose the PXE service to untrusted networks

- **Require a password to PXE boot**

  When you require a password for PXE boot, this configuration adds an extra level of security to the PXE boot process, to help safeguard against rogue clients joining the Configuration Manager hierarchy.

- **Do not include line of business applications or software that contains sensitive data into an image that will be used for PXE boot or multicast**

  Because of the inherent security risks involved with PXE boot and multicast, reduce the risks if rogue computer downloads the operating system image.

- **Do not include line of business applications or software that contains sensitive data in software packages that are installed by using task sequences variables**

  When you deploy software packages by using task sequences variables, software might be installed on computers and to users who are not authorized to receive that software.

- **When you migrate user state, secure the network channel between the client and the state migration point by using SMB signing or IPsec**

  After the initial connection over HTTP, user state migration data is transferred by using SMB. If you do not secure the network channel, an attacker can read and modify this data.

- **Use the latest version of the User State Migration Tool (USMT) that Configuration Manager supports**

  The latest version of USMT provides security enhancements and greater control for when you migrate user state data.

- **Manually delete folders on state migration point when they are decommissioned**

  When you remove a state migration point folder in the Configuration Manager console on the state

migration point properties, the physical folder is not deleted. To protect the user state migration data from information disclosure, you must manually remove the network share and delete the folder.

- **Do not configure the deletion policy to delete user state immediately**

  If you configure the deletion policy on the state migration point to remove data that is marked for deletion immediately, and if an attacker manages to retrieve the user state data before the valid computer does, the user state data would be deleted immediately. Set the **Delete after** interval to be long enough to verify the successful restore of user state data.

- **Manually delete computer associations when the user state migration data restore is complete and verified**

  Configuration Manager does not automatically remove computer associations. Help to protect the identify of user state data by manually deleting computer associations that are no longer required.

- **Manually back up the user state migration data on the state migration point**

  Configuration Manager Backup does not include the user state migration data.

- **Remember to enable BitLocker after the operating system is installed**

  If a computer supports BitLocker, you must disable it by using a task sequence step if you want to install the operating system unattended. Configuration Manager does not enable BitLocker after the operating system is installed, so you must manually re-enable BitLocker.

- **Implement access controls to protect the prestaged media**

  Control physical access to the media to prevent an attacker from using cryptographic attacks to obtain the client authentication certificate and sensitive data.

- **Implement access controls to protect the reference computer imaging process**

  Ensure that the reference computer that you use to capture operating system images is in a secure environment with appropriate access controls so that unexpected or malicious software cannot be installed and inadvertently included in the captured image. When you capture the image, ensure that the destination network file share location is secure so that the image cannot be tampered with after it is captured.

- **Always install the most recent security updates on the reference computer**

  When the reference computer has current security updates, it helps to reduce the window of vulnerability for new computers when they first start up.

- **If you must deploy operating systems to an unknown computer, implement access controls to prevent unauthorized computers from connecting to the network**

  Although provisioning unknown computers provides a convenient method to deploy new computers on demand, it can also allow an attacker to efficiently become a trusted client on your network. Restrict physical access to the network, and monitor clients to detect unauthorized computers. Also, computers responding to PXE-initiated operating system deployment might have all data destroyed during the operating system deployment, which could result in a loss of availability of systems that are inadvertently reformatted.

- **Enable encryption for multicast packages**

  For every operating system deployment package, you have the option to enable encryption when Configuration Manager transfers the package by using multicast. This configuration helps prevent rogue computers from joining the multicast session and helps prevent attackers from tampering with the transmission.

- **Monitor for unauthorized multicast-enabled distribution points**

If attackers can gain access to your network, they can configure rogue multicast servers to spoof operating system deployment.

- **When you export task sequences to a network location, secure the location and secure the network channel**

  Restrict who can access the network folder.

  Use SMB signing or IPsec between the network location and the site server to prevent an attacker from tampering with the exported task sequence.

- **Secure the communication channel when you upload a virtual hard disk to Virtual Machine Manager.**

  To prevent tampering of the data when it is transferred over the network, use Internet Protocol security (IPsec) or server message block (SMB) between the computer that runs the Configuration Manager console and the computer running Virtual Machine Manager.

- **If you must use the Task Sequence Run As Account, take additional security precautions**

  Take the following precautionary steps if you use the Task Sequence Run As Account:

  - Use an account with the least possible permissions.
  - Do not use the Network Access account for this account.
  - Never make the account a domain administrator.

  In addition:

  - Never configure roaming profiles for this account. When the task sequence runs, it will download the roaming profile for the account, which leaves the profile vulnerable to access on the local computer.
  - Limit the scope of the account. For example, create different Task Sequence Run As Accounts for each task sequence, so that if one account is compromised, only the client computers to which that account has access are compromised. If the command line requires administrative access on the computer, consider creating a local administrator account solely for the Task Sequence Run As Account on all computers that will run the task sequence, and delete the account as soon as it is no longer required.

- **Restrict and monitor the administrative users who are granted the Operating System Deployment Manager security role**

  Administrative users who are granted the Operating System Deployment Manager security role can create self-signed certificates that can then be used to impersonate a client and obtain client policy from Configuration Manager.

Security issues for operating system deployment

Although operating system deployment can be a convenient way to deploy the most secure operating systems and configurations for computers on your network, it does have the following security risks:

- Information disclosure and denial of service

  If an attacker can obtain control of your Configuration Manager infrastructure, she could run any task sequences, which might include formatting the hard drives of all client computers. Task sequences can be configured to contain sensitive information, such as accounts that have permissions to join the domain and volume licensing keys.

- Impersonation and elevation of privileges

  Task sequences can join a computer to domain, which can provide a rogue computer with authenticated network access. Another important security consideration for operating system deployment is to protect the

client authentication certificate that is used for bootable task sequence media and for PXE boot deployment. When you capture a client authentication certificate, this gives an attacker an opportunity to obtain the private key in the certificate and then impersonate a valid client on the network.

If an attacker obtains the client certificate that is used for bootable task sequence media and for PXE boot deployment, this certificate can be used to impersonate a valid client to Configuration Manager. In this scenario, the rogue computer can download policy, which can contain sensitive data.

If clients use the Network Access Account to access data stored on the state migration point, these clients effectively share the same identity and could access state migration data from another client that uses the Network Access Account. The data is encrypted so only the original client can read it, but the data could be tampered with or deleted.

- Client authentication to the state migration point is achieved by using a Configuration Manager token that is issued by the management point.

  In addition, Configuration Manager does not limit or manage the amount of data that is stored on the state migration point and an attacker could fill up the available disk space and cause a denial of service.

- If you use collection variables, local administrators can read potentially sensitive information

  Although collection variables offer a flexible method to deploy operating systems, this might result in information disclosure.

Privacy information for operating system deployment

In addition to deploying operating systems to computers with no operating system, Configuration Manager can be used to migrate users' files and settings from one computer to another. The administrator configures which information to transfer, including personal data files, configuration settings, and browser cookies.

The information is stored on a state migration point and is encrypted during transmission and storage. The information is allowed to be retrieved by the new computer associated with the state information. If the new computer loses the key to retrieve the information, a Configuration Manager administrator with the View Recovery Information right on computer association instance objects can access the information and associate it with a new computer. After the new computer restores the state information, it deletes the data after one day by default. You can configure when the state migration point removes data marked for deletion. The state migration information is not stored in the site database and is not sent to Microsoft.

If you use boot media to deploy operating system images, always use the default option to password-protect the boot media. The password encrypts any variables stored in the task sequence, but any information not stored in a variable might be vulnerable to disclosure.

Operating system deployment can use task sequences to perform many different tasks during the deployment process, which includes installing applications and software updates. When you configure task sequences, you should also be aware of the privacy implications of installing software.

If you upload a virtual hard disk to Virtual Machine Manager without first using Sysprep to clean the image, the uploaded virtual hard disk could contain personal data from the original image.

Configuration Manager does not implement operating system deployment by default and requires several configuration steps before you collect user state information or create task sequences or boot images.

Before you configure operating system deployment, consider your privacy requirements.

Planning for operating system deployment interoperability in System Center Configuration Manager

11/23/2016 • 4 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

When different System Center Configuration Manager sites in a single hierarchy use different versions, some Configuration Manager functionality is not available. Typically, functionality from the newer version of Configuration Manager is not accessible at sites or by clients that run a lower version. For more information, see Interoperability between different versions of System Center Configuration Manager.

Consider the following when you upgrade the top-level site in your hierarchy and other sites in your hierarchy run Configuration Manager with a lower version:

- Client installation package
  - The source for the default client installation package is automatically upgraded and all distribution points in the hierarchy are updated with the new client installation package, even on distribution points at sites in the hierarchy that are at a lower version.
  - Clients that run the new version cannot be assigned to sites that have not yet been upgraded to the new version. Assignment is blocked at the management point.
- Boot images
  - When you upgrade the top-level site to the latest version of Configuration Manager, the default boot images (x86 and x64) are automatically updated to Windows ADK for Windows 10-based boot images, which use Windows PE 10. The files that are associated with the default boot images are updated with the latest Configuration Manager version of the files. Custom boot images are not updated automatically. You will need to update custom boot images manually, which includes older Windows PE versions.
  - Avoid the use of dynamic media when your site hierarchy contains sites with different versions of Configuration Manager. Instead, use site-based media to contact a specific management point until all sites are upgraded to the same version of Configuration Manager.
  - Verify that the latest Configuration Manager boot images contain the desired customization, and then update all distribution points at the sites with the latest version of Configuration Manager with the new boot images.
- User State Migration Tool (USMT)
  - When you upgrade the top-level site to the latest version of Configuration Manager, the default USMT package is automatically updated to the latest version. Custom USMT packages are not updated automatically. You will need to manually update these packages.
- New task sequence steps
  - Periodically, new task sequence steps are introduced with new versions of Configuration Manager. When you deploy a task sequence with a new step to older clients, the task sequence step will fail. Before you deploy a task sequence with a new step, make sure the clients in the target collection are updated to the new version.

- Operating system deployment media
  - All media (bootable, capture, prestaged, and stand-alone) must be updated with the new Configuration Manager client package when the site is updated to a new version.
- Third-party extensions to operating system deployment
  - When you have third-party extensions to operating system deployment and you have different versions of Configuration Manager sites or Configuration Manager clients, a mixed hierarchy, there might be issues with the extensions.

While you are actively upgrading sites in your hierarchy, use the following sections to help you with operating system deployments.

Latest version of Configuration Manager sites in a mixed hierarchy

When you upgrade a site to latest version of Configuration Manager, task sequences that reference the default client installation package will automatically start to deploy the latest Configuration Manager client version. Task sequences that reference a custom client installation package will continue to deploy the version of the client that is contained in that custom package (likely a previous Configuration Manager client version), and must be updated to avoid task sequence deployment failures. When you have a task sequence that is configured to use a custom client installation package, you must update the task sequence step to use the latest Configuration Manager version of the client installation package or update the custom package to use the latest Configuration Manager client installation source.

> **IMPORTANT**
> Do not deploy a task sequence that references the latest Configuration Manager client installation package to clients in an older Configuration Manager site. When clients assigned to an older Configuration Manager site are upgraded to the latest Configuration Manager client version, Configuration Manager blocks the assignment to the older Configuration Manager site. Therefore, the client is no longer assigned to any site and will be unmanaged until you manually assign the client to latest Configuration Manager site or reinstall the older Configuration Manager version of the client on the computer.

Older versions of Configuration Manager in a Mixed Hierarchy

When you have upgraded your central administration site to the latest version of Configuration Manager, you must take the following step to ensure that operating system deployment task sequences that you deploy to clients assigned to an older Configuration Manager site (not yet upgraded to the latest version of Configuration Manager) do not leave those clients in an unmanaged state.

- Create a task sequence that you will use to deploy to clients only in a Configuration Manager site. Likely, you will make a copy of a task sequence that you use to deploy to clients in the latest version of Configuration Manager site and then modify the task sequence so you can deploy it to clients in an older Configuration Manager site. Then, configure the task sequence to reference a custom client installation package that uses the older Configuration Manager client installation source. If you do not already have a custom client installation package that references the older Configuration Manager client installation source then you must manually create one.

Prepare site system roles for operating system deployments with System Center Configuration Manager

11/23/2016 • 13 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

To deploy operating systems in System Center Configuration Manager, you must first prepare the following site system roles that require specific configurations and considerations.

Distribution points

The distribution point site system role contains source files for clients to download, such as application content, software updates, operating system images, and boot images. You can control content distribution by using bandwidth, throttling, and scheduling options.

It is important that you have enough distribution points to support the deployment of operating systems to computers. It's also important that you and plan for the placement of these distribution points in your hierarchy. You will find most of this planning information in Manage content and content infrastructure. However, there are some additional planning considerations for distribution points specific to operating system deployment.

Additional planning considerations for distribution points

The following are additional planning things to consider for distribution points:

- **How can I prevent unwanted operating system deployments?**

  Configuration Manager does not distinguish site servers from other destination computers in a collection. If you deploy a required task sequence to a collection that contains a site server, the site server runs the task sequence in the same way that any other computer in the collection runs the task sequence. Ensure that your operating system deployment uses a collection that contains the clients that you intend to run the deployment.

  You can manage the behavior for high-risk task sequence deployments. A high-risk deployment automatically installs on a client and has the potential to cause unwanted results. For example, a task sequence with a purpose of Required that deploys an operating system. To reduce the risk of an unwanted high-risk deployment, you can configure deployment verification settings. For more information, see Settings to manage high-risk deployments.

- **How many computers can receive an operating system image at one time from a single distribution point?**

  To estimate how many distribution points you need, consider the processing speed and disk I/O of the distribution point, the available bandwidth on the network, and the effect that the size of the image package has on these resources. For example, on a 100 megabyte (MB) Ethernet network, the maximum number of computers that can process a 4 gigabyte (GB) image package in one hour is 11 computers if you do not consider any other server resource factors.

  ```
  100 Megabits/sec = 12.5 Megabytes/sec = 750 Megabytes/min = 45 Gigabytes/hour = 11 images @ 4GB per
  image.
  ```

  If you must deploy an operating system to a specific number of computers within a specific time frame, distribute the image to an appropriate number of distribution points.

- **Can I deploy an operating system to a distribution point?**

  You can deploy an operating system to a distribution point, but the operating system image must be received from a different distribution point.

Configuring distribution points to accept PXE requests

To deploy operating systems to Configuration Manager clients that make PXE boot requests, you must configure one or more distribution points to accept PXE requests. Once you configure the distribution point, it will respond to PXE boot request and determine the appropriate deployment action to take.

> **IMPORTANT**
> Windows Deployment Services must be installed on the all PXE-enabled distribution points.

Use the following procedure to modify an existing distribution point so that it can accept PXE requests. For information about how to install a new distribution point, see Install or modify a distribution point.

To modify an existing distribution point to accept PXE requests

1. In the Configuration Manager console, click **Administration**, expand **Overview** and click **Distribution points**.
2. Select the distribution point to configure, and on the **Home** tab in the **Properties** group, click **Properties**.
3. On the property page for the distribution point, click the **PXE** tab. and select **Enable PXE support for clients** to enable PXE on this distribution point.
4. Click **Yes** in the **Review Required Ports for PXE** dialog box to confirm that you want to enable PXE. Configuration Manager automatically configures the default ports on a Windows firewall. You must manually configure the ports if you use a different firewall.

   > **NOTE**
   > If WDS and DHCP are installed on the same server, you must configure WDS to listen on a different port (since DHCP listens on the same port). For more information, see Considerations when you have WDS and DHCP on the same server.

5. Select **Allow this distribution point to respond to incoming PXE requests** to enable WDS so that it responds to incoming PXE service requests. You can use this setting to enable and disable the service without removing the PXE functionality from the distribution point.
6. Select **Enable unknown computer support** to deploy operating systems to computers that are not managed by Configuration Manager.
7. Select **Require a password when computers use PXE**, and then specify a strong password to provide additional security for your PXE deployment.
8. In the **User Device Affinity** list, choose how you want the distribution point to associate users with the destination computer for PXE deployments.
   - Select **Do not use user device affinity** to not associate users with the destination computer.
   - Select **Allow user device affinity with manual approval** to wait for approval from an administrative user before users are associated with the destination computer.
   - Select **Allow user device affinity with automatic approval** to automatically associate users with the destination computer without waiting for approval.

For more information, see Associate users with a destination computer.

9. Specify that the distribution point responds to PXE requests from all network interfaces or from specific network interfaces. If you choose to have the distribution point respond to a specific network interfaces, provide the MAC address for each network interface.

10. Specify, in seconds, how long the delay is for the distribution point before it responds to computer requests when multiple PXE-enabled distribution points are used.

11. Click **OK** to update the properties of the distribution point.

Customize the RamDisk TFTP block size and window size on PXE-enabled distribution points

You can customize the RamDisk TFTP block size, and beginning in Configuration Manager version 1606, the window size for PXE-enabled distribution points. If you have customized your network, it could cause the boot image download to fail with a time-out error because the block or window size is too large. The RamDisk TFTP block size and window size customization allow you to optimize TFTP traffic when using PXE to meet your specific network requirements.
You will need to test the customized settings in your environment to determine what is most efficient.

- **TFTP block size**: The block size is the size of the data packets that are sent by the server to the client that is downloading the file (as discussed in RFC 2347). A larger block size allows the server to send fewer packets, so there are fewer round-trip delays between the server and the client. However, a large block sizes leads to fragmented packets, which most PXE client implementations do not support.

- **TFTP window size**: TFTP requires an acknowledgment (ACK) packet for each block of data that is sent. The server does not send the next block in the sequence until it receives the ACK packet for the previous block. TFTP windowing is a feature in Windows Deployment Services that enables you to define how many data blocks it takes to fill a window. The server sends the data blocks back-to-back until the window is filled, and then the client sends an ACK packet. Increasing this window size reduces the number of round-trip delays between the client and server and decreases the overall time that is required to download a boot image.

To modify the RamDisk TFTP window size

- Add the following registry key on PXE-enabled distribution points to customize the RamDisk TFTP window size:

  Location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SMS\DP
  Name: RamDiskTFTPWindowSize

  Type: REG_DWORD

  Value:

The default value is 1 (1 data block fills the window)

To modify the RamDisk TFTP block size

- Add the following registry key on PXE-enabled distribution points to customize the RamDisk TFTP window size:

  Location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SMS\DP
  Name: RamDiskTFTPBlockSize

  Type: REG_DWORD

  Value:

The default value is 4096 (4k).

Configure distribution points to support multicast

Multicast is a network optimization method that you can use on distribution points when multiple clients are likely

to download the same operating system image at the same time. When multicast is used, multiple computers can simultaneously download the operating system image as it is multicast by the distribution point, rather than having the distribution point send a copy of the data to each client over a separate connection. You must configure at least one distribution point to support multicast. For more information, see Use multicast to deploy Windows over the network.

Before you deploy the operating system, you must configure a distribution point to support multicast. Use the following procedure to modify an existing distribution point to support multicast. For information about how to install a new distribution point, see Install and configure distribution points.

To enable multicast for a distribution point

1. In the Configuration Manager console, click **Administration**.
2. In the **Administration** workspace, expand **Overview**, and then select the **Distribution Points** node.
3. Select the distribution point that you want to use to multicast the operating system image.
4. On the **Home** tab, in the **Properties** group, click **Properties**.
5. Select the **Multicast** tab, and configure the following options:

   - **Enable Multicast**: You must select this option for the distribution point to support multicast.
   - **Multicast Connection Account**: Specify an account to connect to the site database.
   - **Multicast address settings**: Specify the IP addresses to send data to the destination computers. By default, the IP address is obtained from a DHCP server that is enabled to distribute multicast addresses. Depending on the network environment, you can specify a range of IP addresses between 239.0.0.0 and 239.255.255.255.

     > **IMPORTANT**
     > These IP addresses must be accessible by the destination computers that request the operating system image. This means that routers and firewalls in between the destination computer and the site server must be configured to allow multicast traffic.

   - **UDP Port Range**: Specify the range of UDP ports to send data to the destination computers.

     > **IMPORTANT**
     > These ports must be accessible by the destination computers that request the operating system image. This means that routers and firewalls in between the destination computer and the site server must be configured to allow multicast traffic.

   - **Enabled scheduled multicast**: Specify how Configuration Manager controls when to start deploying operating systems to destination computers. Click **Enabled scheduled multicast**, and then select the following options.

     In the **Session start delay** box, specify how many minutes that Configuration Manager waits before it responds to the first deployment request.

     In the **Minimum session size** box, specify how many requests must be received before Configuration Manager starts to deploy the operating system.

   - **Transfer rate**: Select the transfer rate to download data to the destination computers.
   - **Maximum clients**: Specify the maximum number of destination computers that can download the operating system from this distribution point.

6. Click **OK**.

State migration point

The state migration point stores user state data that is captured on one computer and then restored on another computer. However, when you capture user settings for an operating system deployment on the same computer, such as a deployment where you refresh the operating system on the destination computer, you can choose whether to store the data on the same computer by using hard-links or use a state migration point. For some computer deployments, when you create the state store, Configuration Manager automatically creates an association between the state store and the destination computer. As you plan for the state migration point, consider the following factors.

User state size

The size of the user state directly affects disk storage on the state migration point and network performance during the migration. Consider the size of the user state and the number of computers to migrate. Consider also what settings to migrate from the computer. For example, if **My Documents** is already backed up to a server, then perhaps you do not have to migrate it as part of the image deployment. Avoiding unnecessary migrations can keep the overall size of the user state smaller and decrease the effect it would otherwise have on network performance and disk storage on the state migration point.

User State Migration Tool

To capture and restore the user state during the deployment of the operating systems, you must use a User State Migration Tool (USMT) package that points to the USMT source files. Configuration Manager automatically creates this package in the Configuration Manager console in **Software Library** > **Application Management** > **Packages**. Configuration Manager uses USMT 10.0, which is distributed in the Windows Assessment and Deployment Kit (Windows ADK), to capture the user state from one operating system and then restore it to another operating system.

For a description of different migration scenarios for USMT 10.0, see Common Migration Scenarios.

Retention policy

When you configure the state migration point, you can specify the length of time to keep the user state data that is stored on it. The length of time to keep the data on the state migration point depends on two considerations:

- The effect that the stored data has on disk storage.
- The potential requirement to keep the data for a time in case you must migrate the data again.

State migration occurs in two phases: Capturing the data and restoring the data. When you capture data, the user state data is collected and saved to the state migration point. When you restore the data, the user state data is retrieved from the state migration point, written to the destination computer, and then the **Release State Store** task sequence step releases the stored data. When the data is released, the retention timer starts. If you select the option to delete migrated data immediately, the user state data is deleted as soon as it is released. If you select the option to keep the data for a certain period of time, the data is deleted when that period of time elapses after the state data is released. The longer you set the retention period, the more disk space you are likely to require.

Select drive to store user state migration data

When you configure the state migration point, you must specify the drive on the server to store the user state migration data. You select a drive from a fixed list of drives. However, some of these drives might represent non-writable drives, such as the CD drive, or a non-network share drive. In addition, some drive letters might not be mapped to any drives on the computer. You must specify a writable, shared drive when you configure the state migration point.

Configure a state migration point

You can use the following methods to configure a state migration point to store the user state data:

- Use the **Create Site System Server Wizard** to create a new site system server for the state migration point.
- Use the **Add Site System Roles Wizard** to add a state migration point to an existing server.

When you use these wizards, you are prompted to provide the following information for the state migration point:

- The folders to store the user state data.
- The maximum number of clients that can store data on the state migration point.
- The minimum free space for the state migration point to store user state data.
- The deletion policy for the role. You can specify that the user state data is deleted immediately after it is restored on a computer, or after a specific number of days after the user data is restored on a computer.
- Whether the state migration point responds only to requests to restore user state data. When you enable this option, you cannot use the state migration point to store user state data.

For the steps to install a site system role, see Add site system roles.

Prepare for operating system deployment in System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

There are several things you must do in Configuration Manager before you can deploy operating systems. Use the following topics to prepare for operating system deployment:

- Manage boot images
- Manage operating system images
- Manage operating system upgrade packages
- Manage drivers
- Manage user state
- Prepare for unknown computer deployments
- Associate users with a destination computer

Manage boot images with System Center Configuration Manager

2/21/2017 • 14 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

A boot image in Configuration Manager is a Windows PE (WinPE) image that is used during an operating system deployment. Boot images are used to start a computer in WinPE, which is a minimal operating system with limited components and services that prepare the destination computer for Windows installation. Use the following sections to manage boot images.

Default boot images

Configuration Manager provides two default boot images: One to support x86 platforms and one to support x64 platforms. These images are stored in: *servername*>\SMS_<*sitecode*>\osd\boot\<*x64*> or <*i386*>.

When you upgrade Configuration Manager to a new version, Configuration Manager might replace the default boot images, and customized boot images based on the default boot images, in this location with updated files. The options that you configure on the default boot images at the site (such as optional components) are carried forward when the boot images are updated, including drivers. The source driver objects must be valid, including the driver source files, or the drivers will not be added to the updated boot images on the site. Other boot images that are not based on the default boot images, even if based on the same Windows ADK version, will not be updated. After boot images are updated, you will need to redistribute them to distribution points. Any media using the boot images will need to be recreated. If you do not want your customized/default boot images automatically updated, you should store them in a different location.

The Configuration Manager Trace Log Tool is added to all boot images that you add to the **Software Library**. When you are in WinPE, you can start the Configuration Manager Trace Log Tool by typing **CMTrace** from a command prompt.

Customize a boot image

You can customize a boot image, or Modify a boot image, from the Configuration Manager console when it is based on a Windows PE version from the supported version of Windows ADK. When a site is upgraded with a new version and a new version of Windows ADK is installed, custom boot images (not in the default boot image location) are not updated with the new version of Windows ADK. When that happens, you will no longer be able to customize the boot images in the Configuration Manager console. However, they will continue to work as they did before the upgrade.

When a boot image is based on a different version of the Windows ADK installed on a site, you must customize the boot images by using another method, such as using the Deployment Image Servicing and Management (DISM) command-line tool that is part of the Windows AIK and Windows ADK. For more information, see Customize boot images.

Add a boot image

During site installation, Configuration Manager automatically adds boot images that are based on a WinPE version from the supported version of the Windows ADK. Depending on the version of Configuration Manager, you might be able to add boot images based on a different WinPE version from the supported version the Windows ADK. An error occurs when you try to add a boot image that contains an unsupported version of

WinPE.

The following provides the supported version of Windows ADK, the Windows PE version on which the boot image is based that can be customized from the Configuration Manager console, and the Windows PE versions on which the boot image is based that you can customize by using DISM and then add the image to Configuration Manager.

- **Windows ADK version**

  Windows ADK for Windows 10

- **Windows PE versions for boot images customizable from the Configuration Manager console**

  Windows PE 10

- **Supported Windows PE versions for boot images not customizable from the Configuration Manager console**

  Windows PE 3.1[1] and Windows PE 5

  [1] You can only add a boot image to Configuration Manager when it is based on Windows PE 3.1. Install the Windows AIK Supplement for Windows 7 SP1 to upgrade Windows AIK for Windows 7 (based on Windows PE 3) with the Windows AIK Supplement for Windows 7 SP1 (based on Windows PE 3.1). You can download Windows AIK Supplement for Windows 7 SP1 from the Microsoft Download Center.

  For example, when you have Configuration Manager, you can customize boot images from Windows ADK for Windows 10 (based on Windows PE 10) from the Configuration Manager console. However, while boot images based on Windows PE 5 are supported, you must customize them from a different computer and use the version of DISM that is installed with Windows ADK for Windows 8. Then, you can add the boot image to the Configuration Manager console. For more information with the steps to customize a boot image (add optional components and drivers), enable command support to the boot image, add the boot image to the Configuration Manager console, and update distribution points with the boot image, see Customize boot images.

Use the following procedure to manually add a boot image.

To add a boot image

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.
3. On the **Home** tab, in the **Create** group, click **Add Boot Image** to start the Add Boot Image Wizard.
4. On the **Data Source** page, specify the following options, and then click **Next**.
   - In the **Path** box, specify the path to the boot image WIM file.

     The specified path must be a valid network path in the UNC format. For example: \\<*servername*\<*sharename*>\<*bootimagename*>.wim.

   - Select the boot image from the **Boot Image** drop-down list. If the WIM file contains multiple boot images, select the appropriate image.
5. On the **General** page, specify the following options, and then click **Next**.
   - In the **Name** box, specify a unique name for the boot image.
   - In the **Version** box, specify a version number for the boot image.
   - In the **Comment** box, specify a brief description of how the boot image is used.
6. Complete the wizard.

The boot image is now listed in the **Boot Image** node of the Configuration Manager console. However, before you can use the boot image to deploy an operating system you must distribute the boot image to distribution points, distribution point groups, or to collections that are associated with distribution point groups.

> **NOTE**
>
> When you select the **Boot Image** node in the Configuration Manager console, the **Size (KB)** column displays the decompressed size for each boot image. However, when Configuration Manager sends a boot image over the network, it sends a compressed copy of the image, which is typically much smaller than the size listed in the **Size (KB)** column.

Distribute boot images to a distribution point

Boot images are distributed to distribution points in the same way as you distribute other content. In most cases, you must distribute the boot image to at least one distribution point before you deploy an operating system and before you create media.

> **NOTE**
>
> To use PXE to deploy an operating system, consider the following before you distribute the boot image:
>
> - The distribution point must be configured to accept PXE requests.
>   - You must distribute both an x86 and an x64 PXE-enabled boot image to at least one PXE-enabled distribution point.
>   - Configuration Manager distributes the boot images to the **RemoteInstall** folder on the PXE-enabled distribution point.
>
> For more information about using PXE to deploy operating systems, see Use PXE to deploy Windows over the network.

For the steps to distribute a boot image, see Distribute content.

Modify a boot image

You can add or remove device drivers to the image or edit the properties associated with the boot image. The device drivers that you add or remove can include network adapters or mass storage device drivers. Consider the following factors when you modify boot images:

- You must import and enable the device drivers in the device driver catalog before you can add them to the boot image.
- When you modify a boot image, the boot image does not change any of the associated packages that the boot image references.
- After you make changes to a boot image, you must **update** the boot image on the distribution points that already have the boot image so that the most current version of the boot image is available. For more information, see Manage content you have distributed.

Use the following procedure to modify a boot image.

To modify the properties of a boot image

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.
3. Select the boot image that you want to modify.
4. On the **Home** tab, in the **Properties** group, click **Properties** to open the **Properties** dialog box for the boot image.

5. Set any of the following settings to change the behavior of the boot image:

   - On the **Images** tab, if you have changed the properties of the boot image by using an external tool, click **Reload**.
   - On the **Drivers** tab, add the Windows device drivers that are required to boot WinPE. Consider the following when you add device drivers:
     - Select **Hide drivers that do not match the architecture of the boot image** to only display only drivers for the architecture of the boot image. The architecture is based on the architecture reported in the .INF from the manufacturer.
     - Select **Hide drivers that are not in a storage or network class (for boot images)** to only display storage and network drivers, and hide other drivers that are not typically needed for boot images, such as a video driver or modem driver.
     - Select **Hide drivers that are not digitally signed** to hide drivers that are not digitally signed.
     - As a best practice, add only NIC and Mass Storage Drivers to the boot image unless there are requirements for other drivers to be part of WinPE.
     - Because WinPE already comes with many drivers built in, add only NIC and Mass Storage Drivers that are not supplied by WinPE.
     - Make sure that the drivers that you add to the boot image match the architecture of the boot image.

     > **NOTE**
     > You must import device drivers into the drivers catalog before you add them to a boot image. For information about how to import device drivers, see Manage drivers.

   - On the **Customization** tab, select any of the following settings:
     - Select the **Enable Prestart Commands** check box to specify a command to run before the task sequence is run. When prestart commands are enabled, you can then specify the command line that is run, whether support files are required to run the command, and the source location of those support files.

       > **WARNING**
       > You must add **cmd /c** to the start of the command line. If you do not specify **cmd /c**, the command will not close after it runs. The deployment continues to wait for the command to finish and will not start any other configured commands or actions.

       > **TIP**
       > During task sequence media creation, the task sequence writes the package ID and prestart command-line, including the value for any task sequence variables, to the CreateTSMedia.log log file on the computer that runs the Configuration Manager console. You can review this log file to verify the value for the task sequence variables.

     - Set the **Windows PE Background** settings to specify whether you want to use the default WinPE background or a custom background.

- Select **Enable command support (testing only)** to open a command prompt by using the **F8** key while the boot image is deployed. This is useful for troubleshooting while you are testing your deployment. Using this setting in a production deployment is not advised.
- Configure the Windows PE scratch space, which is temporary storage (RAM drive) used by WinPE. For example, when an application is run within WinPE and needs to write temporary files, WinPE redirects the files to the scratch space in memory to simulate the presence of a hard disk. By default, WinPE allocates 32 megabytes (MB) of writeable memory.

- On the **Data Source** tab, update any of the following settings:
  - Set **Image path** and **Image index** to change the source file of the boot image.
  - Select **Update distribution points on a schedule** to create a schedule for when the boot image is updated.
  - Select **Persist content in client cache** if you do not want the content of this package to age out of the client cache to make room for other content.
  - Select **Enable binary differential replication** to specify that only changed files are distributed when the boot image package is updated on the distribution point. This setting minimizes the network traffic between sites, especially when the boot image package is large and the changes are relatively small.
  - Select **Deploy this boot image from the PXE-enabled distribution point** if the boot image is used in a PXE-enabled deployment.

    > **NOTE**
    > For more information, see Use PXE to deploy Windows over the network.

- On the **Data Access** tab, select any of the following settings:
  - Set the **Package share settings** if you want clients to install the content in this package from the network.
  - Set the **Package update settings** to specify how you want Configuration Manager to disconnect users from the distribution point. Configuration Manager might be unable to update the boot image when users are connected to the distribution point.

- On the **Distribution Settings** tab, select any of the following settings:
  - In the **Distribution priority** list, specify the priority level that you want Configuration Manager to use when multiple packages are distributed to the same distribution point.
  - Select **Distribute the content for this package to preferred distribution points** if you want to enable on-demand content distribution to preferred distribution points. When this setting is enabled, the management point distributes the content to all preferred distribution points when a client requests the content for the package and the content is not available on any preferred distribution points.
  - Set the **Prestaged distribution point settings** to specify how you want the boot image to be distributed to distribution points that are enabled for prestaged content.

    > **NOTE**
    > For more information about prestaged content, see Prestage content.

- On the **Content Locations** tab, select the distribution point or distribution point group and perform any of the following actions:
  - Click **Redistribute** to distribute the boot image to the selected distribution point or distribution point group again.
  - Click **Validate** to check the integrity of the boot image package on the selected distribution point or distribution point group.
- On the **Optional Components** tab, specify the components that are added to Windows PE for use with Configuration Manager. For more information about available optional components, see WinPE: Add packages (Optional Components Reference).
- On the **Security** tab, select an administrative user and change the operations that they can perform.

6. After you have configured the properties, click **OK**.

Configure a boot image to deploy from a PXE-enabled distribution point

Before you can use a boot image for a PXE operating system deployment, you must configure the boot image to deploy from a PXE-enabled distribution point.

To configure a boot image to deploy from a PXE-enabled distribution point

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.
3. Select the boot image that you want to modify.
4. On the **Home** tab, in the **Properties** group, click **Properties** to open the **Properties** dialog box for the boot image.
5. On the **Data Source** tab, select **Deploy this boot image from the PXE-enabled distribution point**.

> **NOTE**
> For more information, see Use PXE to deploy Windows over the network.

6. After you have configured the properties, click **OK**.

Configure multiple languages for boot image deployment

Boot images are language neutral. This allows you to use one boot image that will display the task sequence text in multiple languages, while in WinPE, if you include the appropriate language support from the Windows PE Optional Components and set the appropriate task sequence variable to indicate which language can be displayed. The language of the operating system that you deploy is independent from the language that is displayed when in WinPE, regardless of the Configuration Manager version. The language that is displayed to the user is determined as follows:

- When a user runs the task sequence from an existing operating system, Configuration Manager automatically uses the language configured for the user. When the task sequence automatically runs as the result of a mandatory deployment deadline, Configuration Manager uses the language of the operating system.
- For operating system deployments that use PXE or media, you can set the language ID value in the

SMSTSLanguageFolder variable as part of a prestart command. When the computer boots to WinPE, messages are displayed in the language that you specified in the variable. If there is an error accessing the language resource file in the specified folder or you do not set the variable, messages are displayed in the WinPE language.

> **NOTE**
>
> When the media is protected with a password, the text that prompts the user for the password is always displayed in the WinPE language.

Use the following procedure to set the WinPE language for PXE or media-initiated operating system deployments.

To set the Windows PE language for a PXE or media-initiated operating system deployment

1. Verify that the appropriate task sequence resource file (tsres.dll) is in the corresponding language folder on site server before you update the boot image. For example, the English resource file is in the following location: <*ConfigMgrInstallationFolder*>\OSD\bin\x64\00000409\tsres.dll.

2. As part of your prestart command, set the SMSTSLanguageFolder environment variable to the appropriate language ID. The language ID must be specified by using decimal and not hexadecimal. For example, to set the language ID to English, you would specify a decimal value of 1033 instead of the hexadecimal value of 00000409 used for the folder name.

Customize boot images with System Center Configuration Manager

1/23/2017 • 12 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Each version of Configuration Manager supports a specific version of the Windows Assessment and Deployment Kit (Windows ADK). You can service, or customize, boot images from the Configuration Manager console when they are based on a Windows PE version from the supported version of Windows ADK. For other boot images, you must customize them by using another method, such as using the Deployment Image Servicing and Management (DISM) command-line tool that is part of the Windows AIK and Windows ADK.

The following provides the supported version of Windows ADK, the Windows PE version on which the boot image is based that can be customized from the Configuration Manager console, and the Windows PE versions on which the boot image is based that you can customize by using DISM and then add the image to Configuration Manager.

- **Windows ADK version**

  Windows ADK for Windows 10

- **Windows PE versions for boot images customizable from the Configuration Manager console**

  Windows PE 10

- **Supported Windows PE versions for boot images not customizable from the Configuration Manager console**

  Windows PE 3.1[1] and Windows PE 5

  [1] You can only add a boot image to Configuration Manager when it is based on Windows PE 3.1. Install the Windows AIK Supplement for Windows 7 SP1 to upgrade Windows AIK for Windows 7 (based on Windows PE 3) with the Windows AIK Supplement for Windows 7 SP1 (based on Windows PE 3.1). You can download Windows AIK Supplement for Windows 7 SP1 from the Microsoft Download Center.

  For example, when you have Configuration Manager, you can customize boot images from Windows ADK for Windows 10 (based on Windows PE 10) from the Configuration Manager console. However, while boot images based on Windows PE 5 are supported, you must customize them from a different computer and use the version of DISM that is installed with Windows ADK for Windows 8. Then, you can add the boot image to the Configuration Manager console.

The procedures in this topic demonstrate how to add the optional components required by Configuration Manager to the boot image by using the following Windows PE packages:

- **WinPE-WMI**: Adds Windows Management Instrumentation (WMI) support.
- **WinPE-Scripting**: Adds Windows Script Host (WSH) support.
- **WinPE-WDS-Tools**: Installs Windows Deployment Services tools.

There are other Windows PE packages available for you to add. The following resources provide more information about the optional components that you can add to the boot image.

- For Windows PE 5, see WinPE: Add packages (Optional Components Reference)
- For Windows PE 3.1, see the Add a Package to a Windows PE Image topic in the Windows 7 TechNet

Documentation Library.

> **NOTE**
> When you boot to WinPE from a customized boot image that includes tools that you added, you can open a command prompt from WinPE and type the file name of the tool to run it. The location of these tools are automatically added to the path variable. The command prompt can only be added if the **Enable command support (testing only)** setting is selected on the **Customization** tab in the boot image properties.

Customize a boot image that uses Windows PE 5

To customize a boot image that uses Windows PE 5, you must install Windows ADK and use the DISM command-line tool to mount the boot image, add optional components and drivers, and commit the changes to the boot image. Use the following procedure to customize the boot image.

To customize a boot image that uses Windows PE 5

1. Install the Windows ADK on a computer that does not have another version of Windows AIK or Windows ADK, and does not have any Configuration Manager components installed.
2. Download Windows ADK for Windows 8.1 from the Microsoft Download Center.
3. Copy the boot image (wimpe.wim) from the Windows ADK installation folder (for example, *<Installation path>*\Windows Kits*<version>*\Assessment and Deployment Kit\Windows Preinstallation Environment*<x86 or amd64>**<locale>*) to a destination folder on the computer from which you will customize the boot image. This procedure uses C:\WinPEWAIK as the destination folder name.
4. Use DISM to mount the boot image to a local Windows PE folder. For example, type the following command-line:

   dism.exe /mount-wim /wimfile:C:\WinPEWAIK\winpe.wim /index:1 /mountdir:C:\WinPEMount

   Where C:\WinPEWAIK is the folder that contains the boot image and C:\WinPEMount is the mounted folder.

   > **NOTE**
   > For more information about DISM, see the DISM - Deployment Image Servicing and Management Technical Reference topic in the Windows 8.1 and Windows 8 TechNet Documentation Library.

5. After you mount the boot image, use DISM to add optional components to the boot image. In Windows PE 5, the 64-bit optional components are located in *<Installation path>*\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs.

   > **NOTE**
   > This procedure uses the following location for the optional components: C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs. The path you use might be different depending on the version and installation options you choose for the Windows ADK.

   Type the following to install the optional components:

   dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\winpe-wmi.cab"

   dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows

Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\winpe-scripting.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\winpe-wds-tools.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\WinPE-SecureStartup.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\ \WinPE-SecureStartup_ .cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\ \WinPE-WMI_ .cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\ \WinPE-Scripting .cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\ \WinPE-WDS-Tools_ .cab"

Where C:\WinPEMount is the mounted folder and locale is the locale for the components. For example, for the **en-us** locale, you would type:

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\en-us\WinPE-SecureStartup_en-us.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\en-us\WinPE-WMI_en-us.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\en-us\WinPE-Scripting_en-us.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files (x86)\Windows Kits\8.1\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\WinPE_OCs\en-us\WinPE-WDS-Tools_en-us.cab"

> **TIP**
>
> For more information about the optional components that you can add to the boot image, see the Windows PE Optional Components Reference topic in the Windows 8.1 and Windows 8 TechNet Documentation Library.

6. Use DISM to add specific drivers to the boot image, when required. Type the following to add drivers to the boot image:

   dism.exe /image:C:\WinPEMount /add-driver /driver:< *path to driver .inf file* **>**

   Where C:\WinPEMount is the mounted folder.

7. Type the following to unmount the boot image file and commit the changes.

   dism.exe /unmount-wim /mountdir:C:\WinPEMount /commit

   Where C:\WinPEMount is the mounted folder.

8. Add the updated boot image to Configuration Manager to make it available to use in your task sequences. Use the following steps to import the updated boot image:

   a. In the Configuration Manager console, click **Software Library**.

   b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

   c. On the **Home** tab, in the **Create** group, click **Add Boot Image** to start the Add Boot Image Wizard.

   d. On the **Data Source** page, specify the following options, and then click **Next**.

   - In the **Path** box, specify the path to the updated boot image file. The specified path must be a valid network path in the UNC format. For example: **\\<***servername***>\<***WinPEWAIK share***>\winpe.wim**.
   - Select the boot image from the **Boot Image** drop-down list. If the WIM file contains multiple boot images, each image is listed.

   e. On the **General** page, specify the following options, and then click **Next**.

   - In the **Name** box, specify a unique name for the boot image.
   - In the **Version** box, specify a version number for the boot image.
   - In the **Comment** box, specify a brief description of how the boot image is used.

   f. Complete the wizard.

9. You can enable a command shell in the boot image to debug and troubleshoot it in Windows PE. Use the following steps to enable the command shell.

   a. In the Configuration Manager console, click **Software Library**.

   b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

   c. Find the new boot image in the list and identify the package ID for the image. You can find the package ID in the **Image ID** column for the boot image.

   d. From a command prompt, type **wbemtest** to open the Windows Management Instrumentation Tester.

   e. Type **\\<***SMS Provider Computer***>\root\sms\site_<***sitecode***>** in **Namespace**, and then click **Connect**.

   f. Click **Open Instance**, type **sms_bootimagepackage.packageID=""**, and then click **OK**. For packageID, enter the value that you identified in step 3.

   g. Click **Refresh Object**, and then click **EnableLabShell** in the **Properties** pane.

   h. Click **Edit Property**, change the value to **TRUE**, and click **Save Property**.

   i. Click **Save Object**, and then exit the Windows Management Instrumentation Tester.

10. You must distribute the boot image to distribution points, distribution point groups, or to collections that are associated with distribution point groups before you can use the boot image in a task sequence. Use the following steps to distribute the boot image.

a. In the Configuration Manager console, click **Software Library**.

b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

c. Click the boot image identified in step 3.

d. On the **Home** tab, in the **Deployment** group, click **Update Distribution Points**.

Customize a boot image that uses Windows PE 3.1

To customize a boot image that uses WinPE 3.1, you must install Windows AIK, install the Windows AIK supplement for Windows 7 SP1, and use the DISM command-line tool to mount the boot image, add optional components and drivers, and commit the changes to the boot image. Use the following procedure to customize the boot image.

To customize a boot image that uses Windows PE 3.1

1. Install the Windows AIK on a computer that does not have another version of Windows AIK or Windows ADK, and does not have any Configuration Manager components installed. Download Windows AIK from the Microsoft Download Center.

2. Install the Windows AIK Supplement for Windows 7 with SP1 on the computer from step 1. Download Windows AIK Supplement for Windows 7 SP1 from the Microsoft Download Center.

3. Copy the boot image (wimpe.wim) from the Windows AIK installation folder (for example, <*InstallationPath*>\Windows AIK\Tools\PETools\amd64\) to a folder on the computer from which you will customize the boot image. This procedure uses C:\WinPEWAIK as the folder name.

4. Use DISM to mount the boot image to a local Windows PE folder. For example, type the following command-line:

   dism.exe /mount-wim /wimfile:C:\WinPEWAIK\winpe.wim /index:1 /mountdir:C:\WinPEMount

   Where C:\WinPEWAIK is the folder that contains the boot image and C:\WinPEMount is the mounted folder.

   > **NOTE**
   > For more information about DISM, see the Deployment Image Servicing and Management Technical Reference topic in the Windows 7 TechNet Documentation Library.

5. After you mount the boot image, use DISM to add optional components to the boot image. In Windows PE 3.1, for example, the optional components are located in <*InstallationPath*>\Windows AIK\Tools\PETools\amd64\WinPE_FPs\.

   > **NOTE**
   > This procedure uses the following location for the optional components: C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs. The path you use might be different depending on the version and installation options you choose for the Windows AIK.

   Type the following to install the optional components:

   dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\winpe-wmi.cab"

   dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\winpe-scripting.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\winpe-wds-tools.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\ \winpe-wmi_ .cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\ \winpe-scripting_ .cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\ \winpe-wds-tools_ .cab"

Where C:\WinPEMount is the mounted folder and locale is the locale for the components. For example, for the **en-us** locale, you would type:

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\en-us\winpe-wmi_en-us.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\en-us\winpe-scripting_en-us.cab"

dism.exe /image:C:\WinPEMount /add-package /packagepath:"C:\Program Files\Windows AIK\Tools\PETools\amd64\WinPE_FPs\en-us\winpe-wds-tools_en-us.cab"

> **TIP**
> For more information about the different packages that you can add to the boot image, see the Add a Package to a Windows PE Image topic in the Windows 7 TechNet Documentation Library.

6. Use DISM to add specific drivers to the boot image, when required. Type the following to add drivers to the boot image, if required:

   dism.exe /image:C:\WinPEMount /add-driver /driver:< *path to driver .inf file* **>**

   Where C:\WinPEMount is the mounted folder.

7. Type the following to unmount the boot image file and commit the changes.

   dism.exe /unmount-wim /mountdir:C:\WinPEMount /commit

   Where C:\WinPEMount is the mounted folder.

8. Add the updated boot image to Configuration Manager to make it available to use in your task sequences. Use the following steps to import the updated boot image:

   a. In the Configuration Manager console, click **Software Library**.

   b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

   c. On the **Home** tab, in the **Create** group, click **Add Boot Image** to start the Add Boot Image Wizard.

   d. On the **Data Source** page, specify the following options, and then click **Next**.

   - In the **Path** box, specify the path to the updated boot image file. The specified path must be a valid network path in the UNC format. For example: **\\<***servername***>\<***WinPEWAIK share***>\winpe.wim**.
   - Select the boot image from the **Boot Image** drop-down list. If the WIM file contains multiple boot images, each image is listed.

   e. On the **General** page, specify the following options, and then click **Next**.

- In the **Name** box, specify a unique name for the boot image.
- In the **Version** box, specify a version number for the boot image.
- In the **Comment** box, specify a brief description of how the boot image is used.

f. Complete the wizard.

9. You can enable a command shell in the boot image to debug and troubleshoot it in Windows PE. Use the following steps to enable the command shell.

   a. In the Configuration Manager console, click **Software Library**.

   b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

   c. Find the new boot image in the list and identify the package ID for the image. You can find the package ID in the **Image ID** column for the boot image.

   d. From a command prompt, type **wbemtest** to open the Windows Management Instrumentation Tester.

   e. Type **\\<***SMS Provider Computer***>\root\sms\site_<***sitecode***>** in **Namespace**, and then click **Connect**.

   f. Click **Open Instance**, type **sms_bootimagepackage.packageID=""**, and then click **OK**. For packageID, enter the value that you identified in step 3.

   g. Click **Refresh Object**, and then click **EnableLabShell** in the **Properties** pane.

   h. Click **Edit Property**, change the value to **TRUE**, and click **Save Property**.

   i. Click **Save Object**, and then exit the Windows Management Instrumentation Tester.

10. You must distribute the boot image to distribution points, distribution point groups, or to collections that are associated with distribution point groups before you can use the boot image in a task sequence. Use the following steps to distribute the boot image.

    a. In the Configuration Manager console, click **Software Library**.

    b. In the **Software Library** workspace, expand **Operating Systems**, and then click **Boot Images**.

    c. Click the boot image identified in step 3.

    d. On the **Home** tab, in the **Deployment** group, click **Update Distribution Points**.

Manage operating system images with System Center Configuration Manager

12/7/2016 • 7 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Operating system images in Configuration Manager are stored in the Windows Imaging (WIM) file format and represent a compressed collection of reference files and folders that are required to successfully install and configure an operating system on a computer. For all operating system deployment scenarios, you must select an operating system image. You can use the default operating system image or build the operating system image from a reference computer that you configure. When you build the reference computer, you can add operating system files, drivers, support files, software updates, tools, and other software applications to the operating system before you capture it to create the image file. The following provides information about each method.

Default image

The default operating system image (install.wim) is included with the Windows operating system installation files. This image is a basic operating system image that contains a standard set of drivers. When you use the default operating system image, you can install apps and make other configurations after the operating system installs by using task sequence steps. The default operating system image is located in *<operating system source path>*\Sources\install.wim.

- **Advantages**
  - The image size is smaller than a captured image.
  - Installing apps and configurations with task sequence steps is more dynamic. For example, you can change the apps that will install and the configurations in the task sequence and not have to re-image the operating system.
- **Disadvantages**
  - Operating system installation can take more time because the app installation and other configurations occur after the operating system installation completes.

Captured image

To create a customized operating system image, you build a reference computer with the desired operating system, and install apps, configure settings, etc. Then, you capture the operating system image from the reference computer to create the WIM file. You can build the reference computer manually or use a task sequence to automate some or all of the build steps.
For the steps to create a customized operating system image, see Customize operating system images.

- **Advantages**
  - The installation can be faster than using the default image. For example, apps can be pre-installed with the captured operating system image and you won't need to install apps later by using task sequence steps.
- **Disadvantages**
  - Operating system installation can take more time because the app installation and other configurations occur after the operating system installation completes.

Add operating system images to Configuration Manager

Before you can use an operating system image, you must add the image to a Configuration Manager site. Use the following procedure to add an operating system image to a site.

To add an operating system image to a site

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Operating System Images**.
3. On the **Home** tab, in the **Create** group, click **Add Operating System Image** to start the Add Operating System Image Wizard.
4. On the **Data Source** page, specify the network path to the operating system image. For example, specify **\\server\path\OS.WIM**.
5. On the **General** page, specify the following information, and then click **Next**. This information is useful for identification purposes when you add multiple operating system images to the same site.
   - **Name**: Specify the name of the image. By default, the name of the image is taken from the WIM file.
   - **Version**: Specify the version of the image.
   - **Comment**: Specify a brief description of the image.
6. Complete the wizard.

You can now distribute the operating system image to distribution points.

Distribute operating system images to distribution points

Operating system images are distributed to distribution points in the same way as you distribute other content. In most cases, you must distribute the operating system image to at least one distribution point before you deploy the operating system. For the steps to distribute an operating system image, see Distribute content.

Apply software updates to an operating system image

Periodically, new software updates are released that are applicable to the operating system in your operating system image. Before you can apply software updates to an image you must have your software updates infrastructure in place, have successfully synchronized software updates, and downloaded the softare updates to the content library on the site server. For more information, see Deploy software updates.

You can apply applicable software updates to an image on a specified schedule. On the schedule that you specify, Configuration Manager applies the software updates that you select to the operating system image, and then optionally distributes the updated image to distribution points. Information about the operating system image is stored in the site database, including the software updates that were applied at the time of the import. Software updates that have been applied to the image since it was initially added are also stored in the site database. When you start the wizard to apply software updates to the operating system image, the wizard retrieves a list of applicable software updates that have not yet been applied to the image for you to select. Configuration Manager copies the software updates from the content library on the site server and applies the software updates to the operating system image.

Use the following procedure to apply software updates to an operating system image.

To apply software updates to an operating system image

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Operating System**

Images.

3. Select the operating system image to which to apply software updates.
4. On the **Home** tab, in the **Operating System Image** group, click **Schedule Updates** to start the wizard.
5. On the **Choose Updates** page, select the software updates to apply to the operating system image, and then click **Next**.
6. On the **Set Schedule** page, specify the following settings, and then click **Next**.
   a. **Schedule**: Specify the schedule for when the software updates are applied to the operating system image.
   b. **Continue on error**: Select this option to continue to apply software updates to the image even when there is an error.
   c. **Distribute the image to distribution points**: Select this option to update the operating system image on distribution points after the software updates are applied.
7. On the **Summary** page, verify the information, and then click **Next**.
8. On the **Completion** page, verify that the software updates were successfully applied to the operating system image.

Prepare the operating system image for multicast deployments

Use multicast deployments to allow multiple computers to simultaneously download an operating system image. The image is multicast to clients by the distribution point, rather than having the distribution point send a copy of the image to each client over a separate connection. When you choose the Use multicast to deploy Windows over the network operating system deployment method, you must configure the operating system image package to support multicast before you distribute the operating system image to a multicast-enabled distribution point. Use the following procedure to set the multicast options for an existing operating system image package.

To modify an operating system image package to use multicast

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Operating System Images**.
3. Select the operating system image that you want to distribute to the multicast-enabled distribution point.
4. On the **Home** tab, in the **Properties** group, click **Properties**.
5. Select the **Distribution Settings** tab, and configure the following options:

   - **Allow this package to be transferred via multicast (WinPE only)**: You must select this option for Configuration Manager to simultaneously deploy operating system images.
   - **Encrypt multicast packages**: Specify whether the image is encrypted before it is sent to the distribution point. Use this option if the package contains sensitive information. If the image is not encrypted, the contents of the package will be visible in clear text on the network and might be read by an unauthorized user.
   - **Transfer this package only via multicast**: Specify whether you want the distribution point to deploy the image only during a multicast session.

     If you select **Transfer this package only via multicast**, you must also specify **Download content locally when needed by running task sequence** as the deployment option for the operating system image. You can specify the deployment options for the image when you deploy the

operating system image, or you can specify them later by editing the properties of the deployment. The deployment options are on the **Distribution Points** tab of the **Properties** page of the deployment object.

6. Click **OK**.

Customize operating system images with System Center Configuration Manager

1/23/2017 • 5 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Operating system images in System Center Configuration Manager are WIM files and represent a compressed collection of reference files and folders that are required to successfully install and configure an operating system on a computer. A custom operating system image is built and captured from a reference computer that you configure with all the required operating system files, support files, software updates, tools, and other software apps. The extent to which you manually configure the reference computer is up to you. You can completely automate the configuration of the reference computer by using a build and capture task sequence, you can manually configure certain aspects of the reference computer and then automate the rest by using task sequences, or you can manually configure the reference computer without using task sequences. Use the following sections to customize an operating system.

Prepare for the reference computer

There are several things to think about before you use capture an operating system image from a reference computer.

Decide between an automated or manual configuration

The following outlines advantages and disadvantage for an automated and manual configuration of the reference computer.

Automated configuration

Advantages

- The configuration can be completely unattended, which eliminates the requirement for an administrator or user to be present.
- You can reuse the task sequence to repeat the configuration of additional reference computers with a high level of confidence.
- You can modify the task sequence to accommodate differences in reference computers without having to recreate the entire task sequence.

Disadvantages

- The initial action to build a task sequence can take a long time to create and test.
- If the reference computer requirements change significantly, it can take a long time to rebuild and retest the task sequence.

Manual configuration

Advantages

- You do not have to create a task sequence or take the time to test and troubleshoot the task sequence.
- You can install directly from CDs without putting all the software packages (including Windows itself) into a Configuration Manager package.

Disadvantages

- The accuracy of the reference computer configuration depends on the administrator or user who configures the computer.
- You must still verify and test that the reference computer is configured correctly.
- You cannot reuse the configuration method.
- Requires a person to be actively involved throughout the process.

Considerations for the reference computer

The following lists the basic items to consider when you configure a reference computer.

- **Operating system to deploy**

  The reference computer must be installed with the operating system that you intend to deploy to your destination computers. For more information about the operating systems that you can deploy, see Infrastructure requirements for operating system deployment.

- **Appropriate service pack**

  The reference computer must be installed with the operating system that you intend to deploy to your destination computers.

- **Appropriate software updates**

  Install all software applications that you want included in the operating system image that you capture from the reference computer. You can also install software applications when you deploy the captured operating system image to your destination computers.

- **Workgroup membership**

  The reference computer must be configured as a member of a workgroup.

- **Sysprep**

  The System Preparation (Sysprep) tool is a technology that you can use with other deployment tools to install Windows operating systems onto new hardware. Sysprep prepares a computer for disk imaging or delivery to a customer by configuring the computer to create a new computer security identifier (SID) when the computer is restarted. In addition, Sysprep cleans up user and computer-specific settings and data that must not be copied to a destination computer.

  You can manually Sysprep the reference computer by running the following command:

  ```
  Sysprep /quiet /generalize /reboot
  ```

  The /generalize option instructs Sysprep to remove system-specific data from the Windows installation. System-specific information includes event logs, unique security IDs (SIDs), and other unique information. After the unique system information is removed, the computer restarts.

  You can automate Sysprep by using the Prepare Windows for Capture task sequence step or capture media.

  > **IMPORTANT**
  > The Prepare Windows for Capture task sequence step attempts to reset the local administrator password on the reference computer to a blank value before Sysprep runs. If the Local Security policy **Password must meet complexity requirements** is enabled, this task sequence step fails to reset the administrator password. In this scenario, disable this policy before you run the task sequence.

  For more information about Sysprep, see System Preparation (Sysprep) Technical Reference.

- **Appropriate tools and scripts required to mitigate installation scenarios**

  Appropriate tools and scripts required to mitigate installation scenarios

- **Appropriate desktop customization, such as wall paper, branding, and default user profile**

  You can configure the reference computer with the desktop customization properties that you want to include when you capture the operating system image from the reference computer. Desktop properties include wallpaper, organizational branding, and a standard default user profile.

Manually build a reference computer

Use the following procedure to manually build a reference computer.

> **NOTE**
> When you manually build the reference computer, you can capture the operating system image by using capture media. For more information, see Create capture media.

To manually build the reference computer

1. Identify the computer to use as the reference computer.
2. Configure the reference computer with the appropriate operating system and any other software that is required to create the operating system image that you want to deploy.

   > **WARNING**
   > At a minimum, install the appropriate operating system and service pack, support drivers, and required software updates.

3. Configure the reference computer to be a member of a workgroup.
4. Reset the local Administrator password on the reference computer so that the password value is blank.
5. Run Sysprep by using the command: **sysprep /quiet /generalize /reboot**. The /generalize option instructs Sysprep to remove system-specific data from the Windows installation. System-specific information includes event logs, unique security IDs (SIDs), and other unique information. After the unique system information is removed, the computer restarts.

After the reference computer is ready, use a task sequence to capture the operating system image from the reference computer. For detailed steps, see Capture an operating system image from an existing reference computer.

Use a task sequence to build a reference computer

You can automate the process to create a reference computer by using a task sequence to deploy the operating system, drivers, applications, and so on. Use the following steps to build the reference computer and then to capture the operating system image from the reference computer.

- Use a task sequence to build and capture the operating system image from the reference computer. For detailed steps, see Use a task sequence to build and capture a reference computer.

Manage operating system upgrade packages with System Center Configuration Manager

12/7/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

An upgrade package in System Center Configuration Manager contains the Windows Setup source files that are used to upgrade an existing operating system on a computer. Use the following sections manage operating system upgrade packages in Configuration Manager.

Add operating system upgrade packages to Configuration Manager

Before you can use an operating system upgrade package, you must add the package to a Configuration Manager site. Use the following procedure to add an operating system upgrade package to a site.

To add an operating system upgrade package

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Operating System upgrade packages**.
3. On the **Home** tab, in the **Create** group, click **Add Operating System Upgrade Package** to start the Add Operating System Upgrade Wizard.
4. On the **Data Source** page, specify the network path to the installation source files of the operating system upgrade package. For example, specify the UNC **\\server\path** to where the installation source files are located.

   > **NOTE**
   > The installation source files contain Setup.exe and other files and folders to install the operating system.

   > **IMPORTANT**
   > Limit access to the installation source files to prevent unwanted tampering.

5. On the **General** page, specify the following information, and then click **Next**. This information is useful for identification purposes when you have multiple operating system installers.
   - **Name**: Specify the name of the operating system installer.
   - **Version**: Specify the version of the operating system installer.
   - **Comment**: Specify a brief description of the operating system installer.
6. Complete the wizard.

You can now distribute the operating system installer to the distribution points that are accessed by your deployment task sequences.

Distribute operating system images to a distribution point

Operating system images are distributed to distribution points in the same way as you distribute other content. In most cases, you must distribute the operating system image to at least one distribution point before you deploy the operating system. For the steps to distribute an operating system image, see Distribute content.

Apply software updates to an operating system upgrade package

Beginning in Configuration Manager version 1602, you can apply new software updates to the operating system image in your operating system upgrade package. Before you can apply software updates to an upgrade package you must have your software updates infrastructure in place, have successfully synchronized software updates, and downloaded the software updates to the content library on the site server. For more information, see Deploy software updates.

You can apply applicable software updates to an upgrade package on a specified schedule. On the schedule that you specify, Configuration Manager applies the software updates that you select to the operating system upgrade package, and then optionally distributes the updated upgrade package to distribution points. Information about the operating system upgrade package is stored in the site database, including the software updates that were applied at the time of the import. Software updates that have been applied to the upgrade package since it was initially added are also stored in the site database. When you start the wizard to apply software updates to the operating system upgrade package, the wizard retrieves a list of applicable software updates that have not yet been applied to the upgrade package for you to select. Configuration Manager copies the software updates from the content library on the site server and applies the software updates to the operating system upgrade package.

Use the following procedure to apply software updates to an operating system upgrade package.

To apply software updates to an operating system upgrade package

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Operating System upgrade packages**.
3. Select the operating system upgrade package to which to apply software updates.
4. On the **Home** tab, in the **Operating System Upgrade Packages** group, click **Schedule Updates** to start the wizard.
5. On the **Choose Updates** page, select the software updates to apply to the operating system image, and then click **Next**.
6. On the **Set Schedule** page, specify the following settings, and then click **Next**.
    a. **Schedule**: Specify the schedule for when the software updates are applied to the operating system image.
    b. **Continue on error**: Select this option to continue to apply software updates to the image even when there is an error.
    c. **Distribute the image to distribution points**: Select this option to update the operating system image on distribution points after the software updates are applied.
7. On the **Summary** page, verify the information, and then click **Next**.
8. On the **Completion** page, verify that the software updates were successfully applied to the operating system image.

Manage drivers in System Center Configuration Manager

1/23/2017 • 18 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

System Center Configuration Manager provides a driver catalog that you can use to manage the Windows device drivers in your Configuration Manager environment. You can use the driver catalog to import device drivers into Configuration Manager, to group them in packages, and to distribute those packages to distribution points where you can access them when you deploy an operating system. Device drivers can be used when you install the full operating system on the destination computer and when you install Windows PE by using a boot image. Windows device drivers consist of a Setup Information File (INF) file and any additional files that are required to support the device. When an operating system is deployed, Configuration Manager obtains the hardware and platform information for the device from its INF file. Use the following to manage drivers in your Configuration Manager environment.

Device Driver Categories

When you import device drivers, you can assign the device drivers to a category. Device driver categories help group similarly used device drivers together in the driver catalog. For example, you can assign all network adapter device drivers to a specific category. Then, when you create a task sequence that includes the Auto Apply Drivers step, you can specify a specific category of device drivers. Configuration Manager then scans the hardware and selects the applicable drivers from that category to stage on the system for Windows Setup to use.

Driver packages

You can group similar device drivers in packages to help streamline operating system deployments. For example, you might decide to create a driver package for each computer manufacturer on your network. You can create a driver package while you are importing drivers into the driver catalog directly in the **Driver Packages** node. After the driver package is created, it must be distributed to distribution points from which Configuration Manager client computers can install the drivers as they are required. Consider the following:

- When you create a driver package, the source location of the package must point to an empty network share that is not used by another driver package, and the SMS Provider must have Read and Write permissions to that location.
- When you add device drivers to a driver package, Configuration Manager copies the device driver to the driver package source location. You can add only device drivers that have been imported and that are enabled in the driver catalog to a driver package.
- To copy a subset of the device drivers from an existing driver package, create a new driver package, add the subset of device drivers to the new package, and then distribute the new package to a distribution point.

Use the following sections to create and manage driver packages.

Create a driver package

Use the following procedure to create a new driver package.

> **IMPORTANT**
> To create a driver package, you must have an empty network folder that is not used by another driver package. In most cases, you must create a new folder before you start this procedure.

> **NOTE**
> When you use task sequences to install drivers, create driver packages that contain less than 500 device drivers.

Use the following procedure to create a driver package.

To create a driver package

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Driver Packages**.
3. On the **Home** tab, in the **Create** group, click **Create Driver Package**.
4. In the **Name** box, specify a descriptive name for the driver package.
5. In the **Comment** box, enter an optional description for the driver package. Ensure that the description provides information about the contents or the purpose of the driver package.
6. In the **Path** box, specify an empty source folder for the driver package. Enter the path to the source folder in Universal Naming Convention (UNC) format. Each driver package must use a unique folder.

   > **IMPORTANT**
   > The site server account must have **Read** and **Write** permissions to the specified source folder.

The new driver package does not contain any drivers. The next step is to add drivers to the package.

If the **Driver Packages** node contains several packages, you can add folders to the node to separate the packages into logical groups.

Additional actions for driver packages

You can perform additional actions to manage driver packages when you select one or more driver packages from the **Driver Packages** node. These actions include the following:

ACTION	DESCRIPTION
Create Prestage Content file	Creates files that can be used to manually import content and its associated metadata. Use prestaged content when you have low network bandwidth between the site server and the distribution points where the driver package is stored.
Delete	Removes the driver package from the **Driver Packages** node.
Distribute Content	Distributes the driver package to distribution points, distribution point groups, and distribution point groups that are associated with collections.

ACTION	DESCRIPTION
Manage Access Accounts	Adds, modifies, or removes access accounts for the driver package. For more information about Package Access Accounts, see Accounts used in Configuration Manager.
Move	Moves the driver package to another folder in the **Driver Packages** node.
Update Distribution Points	Updates the device driver package on all the distribution points where the package is stored. This action copies only the content that has changed after the last time it was distributed.
Properties	Opens the **Properties** dialog box where you can review and change the content and properties of the device driver. For example, you can change the name and description of the device driver, enable the device driver, and specify on which platforms the device driver can be run.

Device drivers

You can install device drivers on destination computers without including them in the operating system image that is being deployed. Configuration Manager provides a driver catalog that contains references to all the device drivers that you import into Configuration Manager. The driver catalog is located in the **Software Library** workspace and consists of two nodes: **Drivers** and **Driver Packages**. The **Drivers** node lists all the drivers that you have imported into the driver catalog. Use this node to discover the details about each imported driver, modify the drivers in a driver package or boot image, enable or disable a driver, and so on.

Import device drivers into the driver catalog

You must import device drivers into the driver catalog before you can use them when you deploy an operating system. To better manage your device drivers, import only those device drivers that you plan to install as part of your operating system deployment. However, you can also store multiple versions of device drivers in the driver catalog to provide an easy way to upgrade existing device drivers when hardware device requirements change on your network.

As part of the import process for the device driver, Configuration Manager reads the provider, class, version, signature, supported hardware, and supported platform information that is associated with the device. By default, the driver is named after the first hardware device that it supports; however, you can rename the device driver later. The supported platforms list is based on the information in the INF file of the driver. Because the accuracy of this information can vary, manually verify that the device driver is supported after it is imported into the driver catalog.

After you import device drivers into the catalog, you can add the device drivers to driver packages or to boot image packages.

> **IMPORTANT**
> You cannot import device drivers directly into a subfolder of the **Drivers** node. To import a device driver into a subfolder, first import the device driver into the **Drivers** node, and then move the driver to the subfolder.

Use the following procedure to import Windows device drivers.

To import Windows device drivers into the driver catalog

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Drivers**.
3. On the **Home** tab, in the **Create** group, click **Import Driver** to start the **Import New Driver Wizard**.
4. On the **Locate Driver** page, specify the following options, and then click **Next**:

   - **Import all drivers in the following network path (UNC)**: To import all the device drivers that are contained in a specific folder, specify the network path to the device driver folder. For example: **\\servername\folder**.

     > **NOTE**
     > The process to import all drivers can take some time if there are a lot of folders and a lot of driver files (.inf).

   - **Import a specific driver**: To import a specific driver from a folder, specify the network path (UNC) to the Windows device driver .INF or mass storage Txtsetup.oem file of the driver.
   - **Specify the option for duplicate drivers**: Select how you want Configuration Manager to manage driver categories when a duplicate device drive is imported.

   > **IMPORTANT**
   > When you import drivers, the site server must have **Read** permission to the folder, or the import fails.

5. On the **Driver Details** page, specify the following options, and then click **Next**:

   - **Hide drivers that are not in a storage or network class (for boot images)**: Use this setting to only display storage and network drivers, and hide other drivers that are not typically needed for boot images, such as a video driver or modem driver.
   - **Hide drivers that are not digitally signed**: Use this setting to hide drivers that are not digitally signed.
   - In the list of drivers, select the drivers that you want to import into the driver catalog.
   - **Enable these drivers and allow computers to install them**: Select this setting to let computers install the device drivers. By default, this check box is selected.

     > **IMPORTANT**
     > If a device driver is causing a problem or you want to suspend the installation of a device driver, you can disable the device driver by clearing the **Enable these drivers and allow computers to install them** check box. You can also disable drivers after they have been imported.

   - To assign the device drivers to an administrative category for filtering purposes, such as "Desktops" or "Notebooks" categories, click **Categories** and select an existing category or create a new category. You can also use the category assignment to configure which device drivers that are applied to the deployment by the Auto Apply Drivers task sequence step.

6. On the **Add Driver to Packages** page, choose whether to add the drivers to a package and then click **Next**. Consider the following to add the drivers to a package:

   - Select the driver packages that are used to distribute the device drivers.

     Optionally, click **New Package** to create a new driver package. When you create a new driver

package, you must provide a network share that is not in use by other driver packages.

- If the package has already been distributed to distribution points, click **Yes** in the dialog box to update the boot images on distribution points. You cannot use device drivers until they are distributed to distribution points. If you click **No**, you must run the **Update Distribution Point** action before the boot image will contain the updated drivers. If the driver package has never been distributed, you must click **Distribute Content** from the **Driver Packages** node.

7. On the **Add Driver to Boot Images** page, choose whether to add the device drivers to existing boot images, and then click **Next**. If you select a boot image, consider the following:

> **NOTE**
> As a best practice, add only mass storage and network device drivers to the boot images for operating system deployment scenarios.

- Click **Yes** in the dialog box to update the boot images on distribution points. You cannot use device drivers until they are distributed to distribution points. If you click **No**, you must run the **Update Distribution Point** action before the boot image will contain the updated drivers. If the driver package has never been distributed, you must click **Distribute Content** from the **Driver Packages** node.

- Configuration Manager warns you if the architecture for one or more drivers does not match the architecture of the boot images that you selected. If they do not match, click **OK** and go back to the **Driver Details** page to clear the drivers that do not match the architecture of the selected boot image. For example, if you select an x64 and x86 boot image, all drivers must support both architectures. If you select an x64 boot image, all drivers must support the x64 architecture.

  > **NOTE**
  > - The architecture is based on the architecture reported in the .INF from the manufacturer.
  >   - If a driver reports it supports both architectures then you can import it into either boot image.

- Configuration Manager warns you if you add device drivers that are not network or storage drivers to a boot image because in most cases they are not necessary for the boot image. Click **Yes** to add the drivers to the boot image or **No** to go back and modify your driver selection.

- Configuration Manager warns you if one or more of the selected drivers are not properly digitally signed. Click **Yes** to continue and click **No** to go back and make changes to your driver selection.

8. Complete the wizard.

Manage device drivers in a driver package

Use the following procedures to modify driver packages and boot images. To add or remove device drivers, locate the drivers in the **Drivers** node, and then edit the packages or boot images that the selected drivers are associated with.

To modify the device drivers in a driver package

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Drivers**.
3. In the **Drivers** node, select the device drivers that you want to add to the driver package.
4. On the **Home** tab, in the **Driver** group, click **Edit**, and then click **Driver Packages**.

5. To add a device driver, select the check box of the driver packages to which you want to add the device drivers. To remove a device driver, clear the check box of the driver packages from which you want to remove the device driver.

   If you are adding device drivers that are associated with driver packages, you can optionally create a new package, by clicking **New Package**, which opens the **New Driver Package** dialog box.

6. If the package has already been distributed to distribution points, click **Yes** in the dialog box to update the boot images on distribution points. You cannot use device drivers until they are distributed to distribution points. If you click **No**, you must run the **Update Distribution Point** action before the boot image will contain the updated drivers. If the driver package has never been distributed, you must click **Distribute Content** from the **Driver Packages** node. Before the drivers are available, you must update the driver package on distribution points.

   Click **OK**.

Manage device drivers in a boot image

You can add Windows device drivers that have been imported into the driver catalog to boot images. Use the following guidelines when you add device drivers to a boot image:

- Add only mass storage and network adapter device drivers to boot images because other types of drivers are not generally required. Drivers that are not required increase the size of the boot image unnecessarily.
- Add only device drivers for Windows 10 to a boot image because the required version of Windows PE is based on Windows 10.
- Ensure that you use the correct device driver for the architecture of the boot image. Do not add an x86 device driver to an x64 boot image.

Use the following procedure to add or remove device drivers in a boot image.

To modify the device drivers associated with a boot image

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Drivers**.
3. In the **Drivers** node, select the device drivers that you want to add to the driver package.
4. On the **Home** tab, in the **Driver** group, click **Edit**, and then click **Boot images**.
5. To add a device driver, select the check box of the boot image to which you want to add the device drivers. To remove a device driver, clear the check box of the boot image from which you want to remove the device driver.
6. If you do not want to update the distribution points where the boot image is stored, clear the **Update distribution points when finished** check box. By default, the distribution points are updated when the boot image is updated.

   Click **OK** and consider the following:

   - Click **Yes** in the dialog box to update the boot images on distribution points. You cannot use device drivers until they are distributed to distribution points. If you click **No**, you must run the **Update Distribution Point** action before the boot image will contain the updated drivers. If the driver package has never been distributed, you must click **Distribute Content** from the **Driver Packages** node.
   - Configuration Manager warns you if the architecture for one or more drivers does not match the architecture of the boot images that you selected. If they do not match, click **OK** and go back to the **Driver Details** page to clear the drivers that do not match the architecture of the selected boot

image. For example, if you select an x64 and x86 boot image, all drivers must support both architectures. If you select an x64 boot image, all drivers must support the x64 architecture.

> **NOTE**
> - The architecture is based on the architecture reported in the .INF from the manufacturer.
>   - If a driver reports it supports both architectures then you can import it into either boot image.

- Configuration Manager warns you if you add device drivers that are not network or storage drivers to a boot image because in most cases they are not necessary for the boot image. Click **Yes** to add the drivers to the boot image or **No** to go back and modify your driver selection.
- Configuration Manager warns you if one or more of the selected drivers are not properly digitally signed. Click **Yes** to continue and click **No** to go back and make changes to your driver selection.

7. Click **OK**.

Additional actions for device drivers

You can perform additional actions to manage device drivers when you select one or more device drivers from the **Drivers** node. These actions include the following:

ACTION	DESCRIPTION
Categorize	Clears, manages, or sets an administrative category for the selected device drivers.
Delete	Removes the device driver from the **Drivers** node and also removes the driver from the associated distribution points.
Disable	Prohibits the device driver from being installed. You can temporarily disable device drivers so that Configuration Manager client computers and task sequences cannot install them when you are deploying operating systems. **Note:** The Disable action only prevents drivers from installing using the Auto Apply Driver task sequence step.
Enable	Lets Configuration Manager client computers and task sequences install the device driver when the operating system is deployed.
Move	Moves the device driver to another folder in the **Drivers** node.
Properties	Opens the **Properties** dialog box where you can review and change the properties of the device driver. For example, you can change the name and description of the device driver, enable the device driver, and specify which platforms the device driver can be run on.

Use task sequences to install device drivers

Use task sequences to automate how the operating system is deployed. Each step in the task sequence can perform a specific action, such as installing a device driver. You can use the following two task sequence steps to install device drivers while you are deploying operating systems:

- Auto Apply Drivers: This step lets you automatically match and install device drivers as part of an operating

system deployment. You can configure the task sequence step to install only the best matched driver for each detected hardware device, or specify that the task sequence step installs all compatible drivers for each detected hardware device, and then let Windows Setup choose the best driver. In addition, you can specify a category of device drivers to limit the drivers that are available for this step.

- Apply Driver Package: This step lets you make all device drivers in a specific driver package available for Windows Setup. In the specified driver packages, Windows Setup searches for the device drivers that are required. When you create stand-alone media, you must use this step to install device drivers.

When you use these task sequence steps, you can also specify how the device drivers are installed on the computer where you deploy the operating system. For more information, see Manage task sequences to automate tasks.

Use task sequences to install device drivers on computers

Use the following procedure to install device drivers as part of the operating system deployment.

Use a task sequence to install device drivers

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. In the **Task Sequences** node, select the task sequence that you want to modify to install the device driver, and then click **Edit**.
4. Move to the location where you want to add the **Driver** steps, click **Add**, and then select **Drivers**.
5. Add the **Auto Apply Drivers** step if you want the task sequence to install all the device drivers or the specific categories that are specified. Specify the options for the step on the **Properties** tab and any conditions for the step on the **Options** tab.

   Add the **Apply Driver Package** step if you want the task sequence to install only those device drivers from the specified package. Specify the options for the step on the **Properties** tab and any conditions for the step on the **Options** tab.

   > **IMPORTANT**
   > You can select **Disable this step** on the **Options** tab to disable the step to troubleshoot the task sequence.

6. Click **OK** to save the task sequence.

For more information about creating a task sequence to install an operating system, see Create a task sequence to install an operating system.

Driver management reports

You can use several reports in the **Driver Management** reports category to determine general information about the device drivers in the driver catalog. For more information about reports, see Reporting.

Manage user state in System Center Configuration Manager

1/23/2017 • 7 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can use System Center Configuration Manager task sequences to capture and restore the user state data in operating system deployment scenarios where you want to retain the user state of the current operating system. For example:

- Deployments where you want to capture the user state from one computer to restore it on another computer.
- Update deployments where you want to capture and restore the user state on the same computer.

Configuration Manager uses the User State Migration Tool (USMT) 10.0 to manage the migration of user state data from a source computer to a destination computer after the operating system installation completes. For more information about common migration scenarios for the USMT 10.0, see Common Migration Scenarios.

Use the folowing sections to help you capture and restore user data.

Store user state data

When you capture user state, you can store the user state data on the destination computer or on a state migration point. To store the user state on a user state migration point, you must use a Configuration Manager site system server that hosts the state migration point site system role. To store the user state on the destination computer, you must configure your task sequence to store the data locally using links.

> **NOTE**
> The links that are used to store the user state locally are referred to as hard-links. Hard-links is a USMT 10.0 feature that scans the computer for user files and settings and then creates a directory of hard-links to those files. The hard-links are then used to restore the user data after the new operating system is deployed.

> **IMPORTANT**
> You cannot use a state migration point and use hard-links to store the user state data at the same time.

When the user state information is captured, the information can be stored in one of the following ways:

- You can store the user state data remotely by configuring a state migration point. The **Capture** task sequence sends the data to the state migration point. Then, after the operating system is deployed, the **Restore** task sequence retrieves the data and restores the user state on the destination computer.
- You can store the user state data locally to a specific location. In this scenario, the **Capture** task sequence copies the user data to a specific location on the destination computer. Then, after the operating system is deployed, the **Restore** task sequence retrieves the user data from that location.
- You can specify hard links that can be used to restore the user data to its original location. In this scenario, the user state data remains on the drive when the old operating system is removed. Then, after the new operating system is deployed, the **Restore** task sequence uses the hard-links to restore the user state data

to its original location.

Store user data on a state migration point

To store the user state data on a state migration point, you must do the following:

1. Configure a state migration point to store the user state data.
2. Create a computer association between the source computer and the destination computer. You must create this association before you capture the user state on the source computer.
3. Create a task sequence to capture and restore user state in System Center Configuration Manager. Specifically, you must add the following task sequence steps to capture user data from a computer, store the user date on a state migration point, and restore the user data to a computer:
   - Request State Store to request access to a state migration point when capturing state from a computer or restoring state to a computer.
   - Capture User State to capture and store the user state data on the state migration point.
   - Restore User State to restore the user state on the destination computer by retrieving the data from a user state migration point.
   - Release State Store to notify the state migration point that the capture or restore action is complete.

Store user data locally

To store the user state data locally, you must do the following:

- Create a task sequence to capture and restore user state. Specifically, you must add the following task sequence steps to capture user data from a computer and restore the user data to a computer by using hard-links,
  - Capture User State to capture and store the user state data to a local folder by using hard-links.
  - Restore User State to restore the user state on the destination computer by retrieving the data by using hard-links.

    > **NOTE**
    > The user state data that the hard-links reference remains on the computer after the task sequence removes the old operating system. This is the data that is used to restore the user state when the new operating system is deployed.

Configure a state migration point

The state migration point stores user state data that is captured on one computer and then restored on another computer. However, when you capture user settings for an operating system deployment on the same computer, such as a deployment where you refresh the operating system on the destination computer, you can store the data on the same computer by using hard-links or on a state migration point. For some computer deployments, when you create the state store, Configuration Manager automatically creates an association between the state store and the destination computer. You can use the following methods to configure a state migration point to store the user state data:

- Use the **Create Site System Server Wizard** to create a new site system server for the state migration point.
- Use the **Add Site System Roles Wizard** to add a state migration point to an existing server.

When you use these wizards, you are prompted to provide the following information for the state migration point:

- The folders to store the user state data.
- The maximum number of clients that can store data on the state migration point.
- The minimum free space for the state migration point to store user state data.
- The deletion policy for the role. You can specify that the user state data is deleted immediately after it is restored on a computer, or after a specific number of days after the user data is restored on a computer.
- Whether the state migration point responds only to requests to restore user state data. When you enable this option, you cannot use the state migration point to store user state data.

For more information about the state migration point and the steps to configure it, see State migration point.

Create a computer association

Create a computer association to define a relationship between a source computer and a destination computer when you install an operating system on new hardware and want to capture and restore user data settings. The source computer is an existing computer that Configuration Manager manages. When you deploy the new operating system to the destination computer, the source computer contains the user state that is migrated to the destination computer.

> **NOTE**
> It is not supported to create a computer association between computers located in a Configuration Manager parent site with computers located in a child site. Computer Associations are site specific and do not replicate.

To create a computer association

1. In the Configuration Manager console, click **Assets and Compliance**.
2. In the **Assets and Compliance** workspace, click **User State Migration**.
3. On the **Home** tab, in the **Create** group, click **Create Computer Association**.
4. On the **Computer Association** tab of the **Computer Association Properties** dialog box, specify the source computer that has the user state to capture, and the destination computer on which to restore the user state data.
5. On the **User Accounts** tab, specify the user accounts to migrate to the destination computer. Specify one of the following settings:
   - **Capture and restore all user accounts**: This setting captures and restores all user accounts. Use this setting to create multiple associations to the same source computer.
   - **Capture all user accounts and restore specified accounts**: This setting captures all user accounts on the source computer and only restores the accounts that you specify on the destination computer. In addition, you can use this setting when you want to create multiple associations to the same source computer.
   - **Capture and restore specified user accounts**: This setting captures and restores only the accounts that you specify. You cannot create multiple associations to the same source computer when you select this setting.

Restore user state data when an operating system deployment fails

If the operating system deployment fails, use the USMT 10.0 LoadState feature to retrieve the user states data that was captured during the deployment process. This includes data that is stored on a state migration point or data that is saved locally on the destination computer. For more information on this USMT feature, see LoadState Syntax.

Prepare for unknown computer deployments in System Center Configuration Manager

11/23/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Use the information in this topic to deploy operating systems to unknown computers in your System Center Configuration Manager environment. An unknown computer is a computer that is not managed by Configuration Manager. This means that there is no record of these computers in the Configuration Manager database. Unknown computers include the following:

- A computer where the Configuration Manager client is not installed
- A computer that is not imported into Configuration Manager
- A computer that is not been discovered by Configuration Manager

You can deploy operating systems to unknown computers with the following deployment methods:

- Use PXE to deploy Windows over the network
- Use bootable media to deploy an operating system
- Use prestaged media to deploy an operating system

Unknown computer deployment workflow

The following is the basic workflow to deploy an operating system to an unknown computer:

- Select an unknown computer object to use in the deployment. You can deploy the operating system to one of the unknown computer objects in the **All Unknown Computers** collection or you can add the objects in the **All Unknown Computer** collection to another collection. Configuration Manager provides two unknown computer objects in the **All Unknown Computers** collection. One object is for x86 computers and the other object is for x64 computers.

  > **NOTE**
  > The **x86 Unknown Computer** object is for computers that are only x86 capable. The **x64 Unknown Computer** object is for computers that are x86 and x64 capable. In other words, these objects describe the architecture of the destination computer. They do not describe the operating system that you want to deploy to the destination computer.

- Configure a PXE-enabled distribution point or create media to support unknown computer deployments.
- Deploy the task sequence to install the operating system.

Unknown Computer Installation Process

When a computer is first started from PXE or from media, Configuration Manager checks to see if a record for that computer exists in the Configuration Manager database. If there is a record, Configuration Manager then checks to see if there are any task sequences deployed to the record. If there is not a record, Configuration Manager checks to see if there are any task sequences deployed to an unknown computer object. In either case, Configuration

Manager then performs one of the following actions:

- If there is an available task sequence, Configuration Manager prompts the user to run the task sequence.
- If there is a required task sequence, Configuration Manager automatically runs the task sequence.
- If a task sequence is not deployed for the record, Configuration Manager generates an error that there is no deployed task sequence for the destination computer.

When an unknown computer is started, Configuration Manager recognizes the computer as an unprovisioned computer rather than an unknown computer. This means that the computer can now receive the task sequences that were deployed to the unknown computer object. The deployed task sequence then installs an operating system image that must include the Configuration Manager client.

After the Configuration Manager client is installed, a record for the computer is created and the computer is listed in the appropriate Configuration Manager collection. If the computer fails to install the operating system image or the Configuration Manager client, an "Unknown" record for the computer is created and the computer appears in the **All Systems** collection.

> **NOTE**
> During the installation of the operating system image, the task sequence can retrieve collection variables but not computer variables from this computer.

Enabling Unknown Computer Support

Use the following to enable unknown computer support when you deploy an operating system by using PXE, bootable media, and prestaged media.

- **PXE**

  Select the **Enable unknown computer support** check box on the **PXE** tab for a distribution point that is enabled for PXE. For more information, see Configuring distribution points to accept PXE requests.

- **Bootable media**

  Select the **Enable unknown computer support** check box on the **Security** page of the Create Task Sequence Media Wizard. For more information, see Configuring distribution points to accept PXE requests and Use PXE to deploy Windows over the network with System Center Configuration Manager.

- **Prestaged media**

  Select the **Enable unknown computer support** check box on the **Security** page of the Create Task Sequence Media Wizard. For more information, see Create prestaged media with System Center Configuration Manager.

Associate users with a destination computer in System Center Configuration Manager

11/23/2016 • 2 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

When you use System Center Configuration Manager to deploy operating system you can associate users with the destination computer where the operating system is deployed. This configuration includes the following:

- That a single user is the primary user of the destination computer.
- That multiple users are the primary users of the destination computer.

User device affinity supports user-centric management for when you deploy applications. When you associate a user with the destination computer on which to install an operating system, you can later deploy applications to that user and the applications automatically install on the destination computer. However, although you can configure support for user device affinity when you deploy operating systems, you cannot use user device affinity to deploy operating systems.

For more information about user device affinity, see Link users and devices with user device affinity.

How to specify a user when you deploy operating systems

The following table lists the actions that you can take to integrate user device affinity into your operating system deployments. You can integrate user device affinity into PXE deployments, bootable media deployments, and pre-staged media deployments.

ACTION	MORE INFORMATION

ACTION	MORE INFORMATION
Create a task sequence that includes the **SMSTSAssignUsersMode** variable	Add the **SMSTSAssignUsersMode** variable to the beginning of your task sequence by using the Set Task Sequence Variable task sequence step. This variable specifies how the task sequence handles the user information. Set the variable to one of the following values: **Auto**: The task sequence automatically creates a relationship between the user and destination computer and deploys the operating system. **Pending**: The task sequence creates a relationship between the user and the destination computer, but waits for approval from the administrative user before the operating system is deployed. **Disabled**: The task sequence does not associate a user with the destination computer and continues to deploy the operating system. This variable can also be set on a computer or collection. For more information about the built-in variables, see Task sequence built-in variables.
Create a prestart command that gathers the user information	The prestart command can be a Visual Basic (VB) script that has an input box, or it can be an HTML application (HTA) that validates the user data that is entered. The prestart command must set the **SMSTSUdaUsers** variable that is used when the task sequence is run. This variable can be set on a computer, a collection, or a task sequence variable. Use the following format when you add multiple users: *domain\user1, domain\user2, domain\user3*.
Configure how distribution points and media associate the user with the destination computer	When you configure a distribution point to accept PXE boot requests and when you create bootable media or pre-staged media by using the Create Task Sequence Media Wizard, you can specify how the distribution point or media supports associating users with the destination computer where the operating system is deployed. Configuring user device affinity support does not have a built-in method to validate the user identity. This can be important when a technician who is provisioning the computer enters the information on behalf of the user. In addition to setting how the user information is handled by the task sequence, configuring these options on the distribution point and media provides the ability to restrict the deployments that are started from a PXE boot or from a specific type of media.

Prepare Windows PE peer cache to reduce WAN traffic in System Center Configuration Manager

11/23/2016 • 7 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

When you deploy a new operating system in System Center Configuration Manager, computers that run the task sequence can use Windows PE Peer Cache to obtain content from a local peer (a peer cache source) instead of downloading content from a distribution point. This helps minimize wide area network (WAN) traffic in branch office scenarios where there is no local distribution point.

Windows PE Peer Cache is similar to Windows BranchCache, but functions in the Windows Preinstallation Environment (Windows PE). If you start the task sequence from the operating system context, such as from Software Center on the client, Windows PE Peer Cache is not used. The following terms are used to describe the clients that use Windows PE Peer Cache:

- A **peer cache client** is a computer that is configured to use Windows PE Peer Cache.
- A **peer cache source** is a client that is configured for peer cache and that makes content available to other peer cache clients that request that content.

Use the following sections to manage Peer Cache.

Objects stored on a Peer Cache source

A task sequence configured to use Windows PE Peer Cache can get the following content objects while running in Windows PE:

- Operating system image
- Driver package
- Packages and Programs (When the client continues to run the task sequence in the full operating system, the client gets this content from a peer cache source if the task sequence was originally configured for peer cache when running in Windows PE.)
- Additional boot images

The following content objects never transfer using peer cache. Instead, they transfer from a distribution point or by Windows BranchCache if you have configured Windows BranchCache in your environment:

- Applications
- Software updates

How does Windows PE Peer Cache work?

Consider a scenario with a branch office that does not have a distribution point but does have several clients enabled to use Windows PE Peer Cache. You deploy the task sequence configured to use peer cache to several clients that are configured to be part of the peer cache source. The first client to run the task sequence broadcasts a request for a peer with the content. It doesn't find one so it gets the content from a distribution point across the WAN. The client installs the new image and then stores the content in its Configuration Manager client cache so it can function as a peer cache source to other clients. When the next client runs the task sequence, it broadcasts a

request on the subnet for a peer cache source, and that first client responds and makes its cached content available.

Determine what clients will be part of the Windows PE Peer Cache source

To help you determine what computers to select as a Windows PE Peer Cache source, there are several things that you should consider:

- The Windows PE Peer Cache source should be a desktop computer that is always powered on and available to peer cache clients.
- The Windows PE Peer Cache has a client cache size sufficient to store the images.

Requirements for a client to use a Windows PE Peer Cache source

For clients to use a Windows PE Peer Cache source, they must meet the following requirements:

- The Configuration Manager client must be able to communicate across the following ports on your network:
  - Port for the initial network broadcast to find a peer cache source. By default, this is port 8004.
  - Port for content downloading from a peer cache source (HTTP and HTTPS). By default, this port is 8003.

    > **TIP**
    > Clients will use HTTPS to download content when it is available. However, the same port number is used for either HTTP or HTTPS.

- Configure the Client Cache for Configuration Manager Clients on clients to ensure they have enough space to hold and store the images you deploy. Windows PE Peer Cache does not affect the configuration or behavior of the client cache.
- The deployment options for the task sequence deployment must be configured as Download content locally when needed by task sequence.

Configure Windows PE Peer Cache

You can use the following methods to provision a client with peer cache content so it can serve as a peer cache source:

- A peer cache client that cannot find a peer cache source with the content will download it from a distribution point. If the client receives client settings that enable peer cache and the task sequence is configured to preserve the cached content, the client becomes a peer cache source.
- A peer cache client can get content from another peer cache client (a peer cache source). Because the client is configured for peer cache, when it runs a task sequence that is configured to preserve the cached content, the client becomes a peer cache source.
- A client runs a task sequence that includes the optional step, Download Package Content, which is used to prestage the relevant content that is included in the Windows PE Peer Cache task sequence. When you use this method:
  - The client does not need to install the image that is being deployed.
  - In addition to the **Download Package Content** option, the task sequence must also use the **Configuration Manager client cache** option. You use this option to store the content in the clients

cache so the client can act as a peer cache source for other peer cache clients.

The following procedures will help you configure Windows PE Peer Cache on clients and configure task sequences that support peer cache.

To configure the Windows PE Peer Cache source computers

1. In the Configuration Manager console, navigate to **Administration** > **Client Settings**, and then create a new **Custom Client Device Settings** or edit an existing settings object. You can also configure this for the **Default Client Settings** object.

   > **TIP**
   > Use a custom settings object to manage which clients receive this configuration. For example, you might want to avoid configuring this on the laptops of users who are frequently on the move. A highly mobile system can be a poor source to provide content to other peer cache clients.
   >
   > Also remember that when you configure this setting as part of the **Default Client Settings**, the configuration applies to all clients in your environment.

2. Under **Windows PE Peer Cache**, set **Enable Configuration Manager client in full OS to share content** to **Yes**.

   - By default, only HTTP is enabled. If you want to enable clients to download content over HTTPS, set **Enable HTTPS for client peer communication** to **Yes**.
   - By default, the port for broadcasts is set to 8004 and the port for content downloads is set to 8003. You can change both.

3. Save and deploy the Client Settings to the clients that you select to be a peer cache source.

After a device is configured with this settings object, the device is configured to act as a peer cache source. These settings should be deployed to potential peer cache clients to configure the required ports and protocols.

Configure a task sequence for Windows PE Peer Cache

When you configure the task sequence, use the following task sequence variables as Collection Variables on the collection to which the task sequence is deployed:

- **SMSTSPeerDownload**

  Value: TRUE

  This enables the client to use Windows PE Peer Cache.

- **SMSTSPeerRequestPort**

  Value:

  When you do not use the default port configured in the Client Settings (8004), you must configure this variable with a custom value of the network port to use for the initial broadcast.

- **SMSTSPreserveContent**

  Value: TRUE

  This flags the content in the task sequence to be retained in the Configuration Manager client cache after the deployment. This is different than using SMSTSPersisContent which only preserves the content for the duration of the task sequence and uses the task sequence cache, not the Configuration Manager client cache.

For more information, see Task sequence built-in variables.

Validate the success of using Windows PE peer cache

After you use Windows PE peer cache to deploy and install a task sequence, you can confirm that peer cache was successfully used in the process by viewing the **smsts.log** on the client that ran the task sequence.

In the log, locate an entry similar to the following where <*SourceServerName*> identifies the computer from which the client obtained the content. This computer should be a peer cache source, and not a distribution point server. Other details will vary based on your local environment and configurations.

- *<![LOG[Downloaded file from http:// :8003/SCCM_BranchCache$/SS10000C/sccm?/install.wim to C:_SMSTaskSequence\Packages\SS10000C\install.wim]LOG]!>*

Scenarios to deploy enterprise operating systems with System Center Configuration Manager

11/23/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

There following operating system deployment scenarios are available in System Center Configuration Manager:

- Upgrade Windows to the latest version: This scenario upgrades the operating system on computers that currently run Windows 7, Windows 8, Windows 8.1, or Windows 10. The upgrade process retains the applications, settings, and user data on the computer. There are no external dependencies, such as the Windows ADK and this process is faster and more resilient than traditional operating system deployments.
- Refresh an existing computer with a new version of Windows: This scenario partitions and formats (wipes) an existing computer and installs a new operating system on the computer. You can migrate settings and user data after the operating system is installed.
- Install a new version of Windows on a new computer (bare metal): This scenario installs an operating system on a new computer. This is a fresh installation of the operating system and does not include any settings or user data migration.
- Replace an existing computer and transfer settings: This scenario installs an operating system on a new computer. Optionally, you can migrate settings and user data from the old computer to the new computer.

Things to consider before you deploy operating system images

There are certain things that you should consider before you deploy an operating system.

Operating system image size

The size of an operating system image can be quite large. For example, the image size for Windows 7 is 3 gigabytes (GB) or more. The size of the image and the number of computers that you simultaneously deploy the operating system to affects the network performance and available bandwidth. Ensure that you test the network performance to better gauge the affect that the image deployment might have and the time it takes to complete the deployment. Configuration Manager activities that affect network performance include distributing the image to a distribution point, distributing the image from one site to another, and downloading the image to the Configuration Manager client.

Also ensure that you plan for sufficient disk storage space on the distribution points that host the operating system images.

Client cache size

When Configuration Manager clients download content, they automatically use Background Intelligent Transfer Service (BITS) if it is available. When you deploy a task sequence that installs an operating system, you can set an option on the deployment so that Configuration Manager clients download the full image to a local cache before the task sequence runs.

In general, when a Configuration Manager client must download an operating system image (or any other package), but there is not enough space in the cache, the client checks the other packages in the cache to determine whether deleting any, or all, of the oldest packages will free enough disk space to accommodate the image. If deleting packages does not free enough disk space, the client does not download the image and the deployment fails. This might occur if the cache has a large package that is configured to persist in the cache. If

deleting packages does free enough disk space in the cache, the client deletes them, and then downloads the image into the cache.

The default cache size on Configuration Manager clients might not be large enough for most operating system image deployments. If you plan to download the full image to the client cache, you must adjust the Configuration Manager client cache size on the destination computers to accommodate the size of the image that you are deploying.

For more information, see Configure the Client Cache for Configuration Manager Clients.

Task sequence deployments

The task sequence that you create can deploy the operating system image on a Configuration Manager client computer in one of the following ways:

- Download the image and its content first to the Configuration Manager client cache from a distribution point and then install it.
- Install the image and its content immediately from the distribution point.
- Install the image and its content as it is required from the distribution point

By default, when you create the deployment for the task sequence, the image is downloaded first to the Configuration Manager client cache and then installed. If you select to download the image to the Configuration Manager client cache before you run the image, and the task sequence contains a step to repartition the hard drive, the repartition step fails because partitioning the hard drive erases the contents of the Configuration Manager client cache. If the task sequence must repartition the hard drive, you must run the image installation from the distribution point by using the **Run program from distribution point** option when you deploy the task sequence.

For more information, see Deploy a task sequence.

Upgrade Windows to the latest version with System Center Configuration Manager

2/7/2017 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic provides the steps in System Center Configuration Manager to upgrade an operating system on a computer from Windows 7 or later to Windows 10. You can choose from different deployment methods, such as stand-alone media or Software Center. The Windows 10 in-place upgrade scenario:

- Upgrades the operating system on computers that currently run Windows 7, Windows 8, or Windows 8.1. You can also do build-to-build upgrades of Windows 10. For example, you can upgrade Windows 10 RTM to Windows 10, version 1511.
- Retains the applications, settings, and user data on the computer.
- Has no external dependencies, such as the Windows ADK.
- Is generally faster and more resilient than traditional operating system deployments.

Use the following sections to deploy operating systems over the network by using a task sequence.

Plan

- **Review the limitations for the task sequence to upgrade an operating system**

  Review the following requirements and limitations for the task sequence to upgrade an operating system to make sure it meets your needs:

  - You should only add task sequence steps that are related to the core task of deploying operating systems and configuring computers after the image is installed. This includes steps that install packages, applications, or updates, and steps that run command lines, PowerShell, or set dynamic variables.
  - Review drivers and applications that are installed on computers to ensure they are compatible with Windows 10 before you deploy the upgrade task sequence.
  - The following tasks are not compatible with the in-place upgrade and require you to use traditional operation system deployments:
    - Changing the computers domain membership or update Local Administrators.
    - Implementing a fundamental change on the computer, including disk partitioning, a changing an architecture from x86 to x64, implementing UEFI, or modifying the base operating system language.
    - You have custom requirements including using a custom base image, using 3rd party disk encryption, or require WinPE offline operations.

- **Plan for and implement infrastructure requirements**

  The only prerequisites for the upgrade scenario is that you have a distribution point available for the operating system upgrade package and any other packages that you include in the task sequence. For more information, see Install or modify a distribution point.

Configure

1. **Prepare the operating system upgrade package**

   The Windows 10 upgrade package contains the source files necessary to upgrade the operating system on the destination computer. The upgrade package must be the same edition, architecture, and language as the clients that you will upgrade. For more information, see Manage operating system upgrade packages.

2. **Create a task sequence to upgrade the operating system**

   Use the steps in Create a task sequence to upgrade an operating system to automate the upgrade of the operating system.

   > [IMPORTANT] When you use stand-alone media, you must include a boot image in the task sequence for it to be available in the Task Sequence Media Wizard.

   > **NOTE**
   > Typically you will use the steps in Create a task sequence to upgrade an operating system to create a task sequence to upgrade an operating system to Windows 10. The task sequence includes the Upgrade Operating System step, as well as additional recommended steps and groups to handle the end-to-end upgrade process. However, you can create a custom task sequence and add the Upgrade Operating System task sequence step to upgrade the operating system. This is the only step required to upgrade the operating system to Windows 10. If you choose this method, also add the Restart Computer step after the Upgrade Operating System step to complete the upgrade. Be sure to use the **The currently installed default operating system** setting to restart the computer into the installed operating system and not Windows PE.

Deploy

- Use one of the following deployment methods to deploy the operating system:
  - Use Software Center to deploy Windows over the network
  - Use stand-alone media to deploy Windows without using the network

Monitor

- **Monitor the task sequence deployment**

  To monitor the task sequence deployment to upgrade the operating system, see Monitor operating system deployments.

Refresh an existing computer with a new version of Windows using System Center Configuration Manager

11/23/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic provides the general steps in System Center Configuration Manager to partition and format (wipe) an existing computer and install a new operating system on the computer. For this scenario, you can choose from many different deployment methods, such as PXE, bootable media, or Software Center. You can also choose to install a state migration point to store settings and then restore them to the new operating system after it is installed. If you are unsure that this is the right operating system deployment scenario for you, see Scenarios to deploy enterprise operating systems.

Use the following sections to refresh an existing computer with a new version of Windows.

Plan

- **Plan for and implement infrastructure requirements**

  There are several infrastructure requirements that must be in place before you can deploy operating systems, such as Windows ADK, User State Migration Tool (USMT), Windows Deployment Services (WDS), supported hard disk configurations, etc. For more information, see Infrastructure requirements for operating system deployment.

- **Install a state migration point (required only if you transfer settings)**

  When you are going to capture settings from the existing computer, and then restore the settings to the new operating system, you must install a state migration point. For more information, see State migration point.

Configure

1. **Prepare a boot image**

   Boot images start a computer in a Windows PE environment (a minimal operating system with limited components and services) that can then install a full Windows operating system on the computer. When you deploy operating systems, you must select a boot image to use and distribute the image to a distribution point. Use the following to prepare the boot image:

   - To learn more about boot images, see Manage boot images.
   - For more information about how to customize a boot image, see Customize boot images.
   - Distribute the boot image to distribution points. For more information, see Distribute content.

2. **Prepare an operating system image**

   The operating system image contains the files necessary to install the operating system on the destination computer. Use the following to prepare the operating system image:

   - To learn more about how to create an operating system image, see Manage operating system

images.

- Distribute the operating system image to distribution points. For more information, see Distribute content.

3. **Create a task sequence to deploy operating systems over the network**

   Use a task sequence to automate the installation of the operating system over the network. Use the steps in Create a task sequence to install an operating system to create the task sequence to deploy the operating system. Depending on the deployment method that you choose, there might be additional considerations for the task sequence.

   > **NOTE**
   > In this scenario, the task sequence formats and partitions the hard disks on the computer. To capture user settings, you must use the state migration point, and select **Save user settings and files on a State Migration Point** on the **State Migration** page of the Create Task Sequence wizard. If you save the user settings and files locally, they will be lost when the hard disk is formatted and Configuration Manager will be unable to restore the settings. For more information, see Manage user state.

Deploy

- Use one of the following deployment methods to deploy the operating system:
  - Use PXE to deploy Windows over the network
  - Use multicast to deploy Windows over the network
  - Create an image for an OEM in factory or a local depot
  - Use stand-alone media to deploy Windows without using the network
  - Use bootable media to deploy Windows over the network
  - Use Software Center to deploy Windows over the network

Monitor

- **Monitor the task sequence deployment**

  To monitor the task sequence deployment to install the operating system, see Monitor operating system deployments.

Install a new version of Windows on a new computer (bare metal) with System Center Configuration Manager

1/23/2017 • 2 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic provides the general steps in System Center Configuration Manager to install an operating system on a new computer. For this scenario, you can choose from many different deployment methods, such as PXE, OEM, or stand-alone media. If you are unsure that this is the right operating system deployment scenario for you, see Scenarios to deploy enterprise operating systems.

Use the following sections to refresh an existing computer with a new version of Windows.

Plan

- **Plan for and implement infrastructure requirements**

  There are several infrastructure requirements that must be in place before you can deploy operating systems, such as Windows ADK, Windows Deployment Services (WDS), supported hard disk configurations, etc. For more information, see Infrastructure requirements for operating system deployment.

Configure

1. **Prepare a boot image**

   Boot images start a computer in a Windows PE environment (a minimal operating system with limited components and services) that can then install a full Windows operating system on the computer. When you deploy operating systems, you must select a boot image to use and distribute the image to a distribution point. Use the following to prepare the boot image:

   - To learn more about boot images, see Manage boot images.
   - For more information about how to customize a boot image, see Customize boot images.
   - Distribute the boot image to distribution points. For more information, see Distribute content.

2. **Prepare an operating system image**

   The operating system image contains the files necessary to install the operating system on the destination computer. Use the following to prepare the operating system image:

   - To learn more about how to create an operating system image, see Manage operating system images.
   - Distribute the operating system image to distribution points. For more information, see Distribute content.

3. **Create a task sequence to deploy operating systems over the network**

   Use a task sequence to automate the installation of the operating system over the network. Use the steps in Create a task sequence to install an operating system to create the task sequence to deploy the operating

system. Depending on the deployment method that you choose, there might be additional considerations for the task sequence.

Deploy

- Use one of the following deployment methods to deploy the operating system:
  - Use PXE to deploy Windows over the network
  - Use multicast to deploy Windows over the network
  - Create an image for an OEM in factory or a local depot
  - Use stand-alone media to deploy Windows without using the network
  - Use bootable media to deploy Windows over the network

Monitor

- **Monitor the task sequence deployment**

  To monitor the task sequence deployment to install the operating system, see Monitor operating system deployments.

Replace an existing computer and transfer settings with System Center Configuration Manager

11/23/2016 • 2 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic provides the general steps in System Center Configuration Manager to replace an existing computer with a new computer. For this scenario, you can choose from many different deployment methods, such as bootable media, multicast, or Software Center. You can also choose to install a state migration point to store settings and then restore them to the new operating system after it is installed. If you are unsure that this is the right operating system deployment scenario for you, see Scenarios to deploy enterprise operating systems.

Use the following sections to refresh an existing computer with a new version of Windows.

Plan

- **Plan for and implement infrastructure requirements**

  There are several infrastructure requirements that must be in place before you can deploy operating systems, such as Windows ADK, User State Migration Tool (USMT), Windows Deployment Services (WDS), supported hard disk configurations, etc. For more information, see Infrastructure requirements for operating system deployment

- **Install a state migration point (required only if you transfer settings)**

  When you are going to capture settings from the existing computer, and then restore the settings to the new operating system, you must install a state migration point. For more information, see State migration point.

Configure

1. **Prepare a boot image**

   Boot images start a computer in a Windows PE environment (a minimal operating system with limited components and services) that can then install a full Windows operating system on the computer. When you deploy operating systems, you must select a boot image to use and distribute the image to a distribution point. Use the following to prepare the boot image:

   - To learn more about boot images, see Manage boot images.
   - For more information about how to customize a boot image, see Customize boot images.
   - Distribute the boot image to distribution points. For more information, see Distribute content.

2. **Prepare an operating system image**

   The operating system image contains the files necessary to install the operating system on the destination computer. Use the following to prepare the operating system image:

   - To learn more about how to create an operating system image, see Manage operating system images.
   - Distribute the operating system image to distribution points. For more information, see Distribute content.

3. **Create a task sequence to deploy operating systems over the network**

   Use a task sequence to automate the installation of the operating system over the network. Use the steps in Create a task sequence to install an operating system to create the task sequence to deploy the operating system. Depending on the deployment method that you choose, there might be additional considerations for the task sequence.

   > **NOTE**
   > In this scenario, if you capture and restore user settings and files, you can choose to use a state migration point or save the files locally. For more information, see Manage user state.

Deploy

- Use one of the following deployment methods to deploy the operating system:
  - Use Software Center to deploy Windows over the network
  - Use bootable media to deploy Windows over the network
  - Use multicast to deploy Windows over the network
  - Create an image for an OEM in factory or a local depot

Monitor

- **Monitor the task sequence deployment**

  To monitor the task sequence deployment to install the operating system, see Monitor operating system deployments.

Methods to deploy enterprise operating systems using System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

There are different methods that you can use to deploy an operating system in your System Center Configuration Manager environment.

- Use PXE to deploy Windows over the network
- Use Software Center to deploy Windows over the network
- Use bootable media to deploy Windows over the network
- Use multicast to deploy Windows over the network
- Use stand-alone media to deploy Windows without using the network
- Create an image for an OEM in factory or a local depot
- Deploy Windows to Go

Use PXE to deploy Windows over the network with System Center Configuration Manager

12/7/2016 • 6 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

PXE-initiated operating system deployments in System Center Configuration Manager let client computers request and deploy operating systems over the network. In this operating system deployment scenario, the operating system image and both the x86 and x64 Windows PE boot images are sent to a distribution point that is configured to accept PXE boot requests.

> **NOTE**
> When you create an operating system deployment that targets only x64 BIOS computers, both the x64 boot image and x86 boot image must be available on the distribution point.

You can use PXE-initiated operating system deployments in the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)

Complete the steps in one of the operating system deployment scenarios and then use the following sections to prepare for PXE-initiated deployments.

Configure at least one distribution point to accept PXE requests

To deploy operating systems to Configuration Manager clients that make PXE boot requests, you must use one or more distribution points that are configured to respond to the PXE boot requests. For the steps to enable PXE on a distribution point, see Configuring distribution points to accept PXE requests.

Prepare a PXE-enabled boot image

To use PXE to deploy an operating system, you must have both x86 and x64 PXE-enabled boot images distributed to one or more PXE-enabled distribution points. Use the information to enable PXE on a boot image and distribute the boot image to distribution points:

- To enable PXE on a boot image, select **Deploy this boot image from the PXE-enabled distribution point** from the **Data Source** tab in the boot image properties.
- If you change the properties for the boot image, re-distribute the boot image to distribution points. For more information, see Distribute content.

Create an exclusion list for PXE deployments

When you use PXE to deploy operating systems, you can create an exclusion list on each distribution point to ignore PXE boot requests from computers that are in the exclusion list. The exclusion list contains MAC addresses of the computers that you want the distribution point to ignore. These computers will not receive the deployment task sequences that Configuration Manager uses for PXE deployment.

Use the following steps to create the PXE exclusion list.

To create the exclusion list

1. Create a text file on the distribution point that is enabled for PXE. As an example, name this text file **pxeExceptions.txt**.

2. Use a standard text editor, such as Notepad, and add the MAC addresses of the computers to be ignored by the PXE-enabled distribution point. Separate the MAC address values by colons, and enter each address on a separate line. For example: `01:23:45:67:89:ab`

3. Save the text file on the PXE-enabled distribution point site system server. The text file can be saved to any location on the server.

4. Edit the registry of the PXE-enabled distribution point to create a **MACIgnoreListFile** registry key that contains the string value of the full path to the location of the text file on the PXE-enabled distribution point site system server. Use the following registry path:

   HKLM\Software\Microsoft\SMS\DP

   > **WARNING**
   > If you use the Registry Editor incorrectly, you might cause serious problems that might require you to reinstall the operating system. Microsoft cannot guarantee that you can solve problems that result from using the Registry Editor incorrectly. Use the Registry Editor at your own risk.

   There is no need to restart the server after you make this registry change.

RamDisk TFTP block size and window size

You can customize the RamDisk TFTP block size, and beginning in Configuration Manager version 1606, the window size for PXE-enabled distribution points. If you have customized your network, it could cause the boot image download to fail with a time-out error because the block or window size is too large. The RamDisk TFTP block size and window size customization allow you to optimize TFTP traffic when using PXE to meet your specific network requirements. You will need to test the customized settings in your environment to determine what is most efficient. For more information, see Customize the RamDisk TFTP block size and window size on PXE-enabled distribution points.

Configure deployment settings

To use a PXE-initiated operating system deployment, you must configure the deployment to make the operating system available for PXE boot requests. You can configure this on the **Deployment Settings** page of the Deploy Software Wizard or the **Deployment Settings** tab in the properties for the deployment. For the **Make available to the following** setting, configure one of the following:

- Configuration Manager clients, media and PXE
- Only media and PXE
- Only media and PXE (hidden)

Deploy the task sequence

Deploy the operating system to a target collection. For more information, see Deploy a task sequence. When you deploy operating systems by using PXE, you can configure whether the deployment is required or available.

- **Required deployment**: Required deployments will use PXE without any user intervention. The user will not be able to bypass the PXE boot. However, if the user cancels the PXE boot before the distribution point

responds, the operating system will not be deployed.

- **Available deployment**: Available deployments require that the user is present at the destination computer so that they can press the F12 key to continue the PXE boot process. If the user is not present to press F12, the computer will boot into the current operating system or from the next available boot device.

You can re-deploy a required PXE deployment by clearing the status of the last PXE deployment assigned to a Configuration Manager collection or a computer. This action resets the status of that deployment and re-deploys the most recent required deployments.

> **IMPORTANT**
> The PXE protocol is not secure. Ensure that the PXE server and the PXE client are located on a physically secure network, such as in a data center to prevent unauthorized access to your site.

How is the boot image selected for clients booting with PXE?

When a client boots with PXE, Configuration Manager provides the client with a boot image to use. Starting in Configuration Manager version 1606, Configuration Manager uses a boot image with an exact architecture match if one is available. If a boot image with the exact architecture is not available, Configuration Manager uses a boot image with a compatible architecture. The following list provides details about how a boot image is selected for clients booting with PXE.

1. Configuration Manager looks in the site database for the system record that matches the MAC address or SMBIOS of the client that is trying to boot.

   > **NOTE**
   > If a computer that is assigned to a site boots to PXE for a differenet site, the policies are not visible for the computer. For example, if a client is already assigned to site A, the management point and distribution point on site B will not be able to access the policies from site A and the client will not successfully PXE boot.

2. Configuration Manager looks for task sequences that are deployed to the system record found in step 1.
3. In the list of task sequences found in step 2, Configuration Manager looks for a boot image that matches the architecture of the client that is trying to boot. If a boot image is found with the same architecture, that boot image is used.
4. If a boot image is not found with the same architecuture, Configuration Manager looks for a boot image (from the list of task sequences found in step 2) that is compatible with the architecture of the client that is trying to boot. For example, a 64-bit client is compatible with 32-bit and 64-bit boot images. A 32-bit client is compatible with only 32-bit boot images. A UEFI client is compatible with only 64-bit boot images.

Use Software Center to deploy Windows over the network with System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

The task sequence to install an operating system in System Center Configuration Manager can be made available in Software Center. You can deploy an operating system to Software Center in the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Upgrade Windows to the latest version

Complete the steps in one of the operating system deployment scenarios and then use the following sections to prepare for deployments that are available in Software Center.

Configure deployment settings

When you want the operating system deployment to be available in the Software Center, you must configure the deployment to make the operating system available to Configuration Manager clients. You can configure this on the **Deployment Settings** page of the Deploy Software Wizard or the **Deployment Settings** tab in the properties for the deployment. For the **Make available to the following** setting, configure either **Only Configuration Manager Clients** or **Configuration Manager clients, media and PXE**. After the operating system is deployed, it will be displayed in Software Center for members of the target collection.

Deploy the task sequence to computers

Deploy the operating system to a target collection. For more information, see Deploy a task sequence. When you deploy operating systems for Software Center, you can configure whether the deployment is required or available.

- **Required deployment**: Required deployments will make the operating system available in Software Center, but it will be automatically started at the configured assignment schedule.
- **Available deployment**: The operating system will be available in the Software Center and the user can install it on demand.

Use bootable media to deploy Windows over the network with System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Bootable media deployments in System Center Configuration Manager let you deploy the operating system when the destination computer starts. When the destination computer starts, it retrieves the task sequence, the operating system image, and any other required content from the network. Because that content is not included on the media, you can update the content without having to re-create the media.

You can deploy operating systems over the network by using multicast in the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)
- Replace an existing computer and transfer settings

Complete the steps in one of the operating system deployment scenarios and then use the following sections to use bootable media to deploy the operating system.

Configure deployment settings

When you use bootable media to start the operating system deployment process, you must configure the deployment to make the operating system available to media. You can configure this on the **Deployment Settings** page of the Deploy Software Wizard or the **Deployment Settings** tab in the properties for the deployment. For the **Make available to the following** setting, configure one of the following:

- **Configuration Manager clients, media and PXE**
- **Only media and PXE**
- **Only media and PXE (hidden)**

Create the bootable media

You can specify whether the bootable media is a USB flash drive or CD/DVD set. The computer that will start the media must support the option that you choose as a bootable drive. For more information, see Create bootable media.

Install the operating system from bootable media

Insert the bootable media in a bootable drive on the computer, and then power it on to install the operating system.

Use stand-alone media to deploy Windows without using the network in System Center Configuration Manager

11/23/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Stand-alone media in System Center Configuration Manager contains everything that is required to deploy an operating system on a computer. This includes the boot image, operating system image, and task sequence to install the operating system, including applications, drivers, and so on. Stand-alone media deployments let you deploy operating systems in the following conditions:

- In environments where it is not practical to copy an operating system image or other large packages over the network.
- In environments without network connectivity or low bandwidth network connectivity.

You can use stand-alone media in the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)
- Upgrade Windows to the latest version

Complete the steps in one of the operating system deployment scenarios and then use the following sections to prepare for and create the stand-alone media.

Task sequence actions not supported when using stand-alone media

If you have completed the steps in one of the supported operating system deployment scenarios, the task sequence to deploy, or upgrade, the operating system has been created and all associated content has been distributed to a distribution point. When you use stand-alone media, the following actions are not supported in the task sequence:

- The Auto Apply Drivers step in the task sequence. Automatic application of device drivers from the driver catalog is not supported, but you can choose the Apply Driver Package step to make a specified set of drivers available to Windows Setup.
- Installing software updates.
- Installing software before deploying the operating system.
- Associating users with the destination computer to support user device affinity.
- Dynamic package installs via the Install Packages task.
- Dynamic application installs via the Install Application task.

NOTE

If your task sequence to deploy an operating system includes the Install Package step and you create the stand-alone media at a central administration site, an error might occur. The central administration site does not have the necessary client configuration policies that are required to enable the software distribution agent during the execution of the task sequence. The following error might appear in the CreateTsMedia.log file:

```
"WMI method SMS_TaskSequencePackage.GetClientConfigPolicies failed (0x80041001)"
```

For stand-alone media that includes an **Install Package** step, you must create the stand-alone media at a primary site that has the software distribution agent enabled or add a Run Command Line step after the Setup Windows and ConfigMgr step and before the first **Install Package** step in the task sequence. The **Run Command Line** step runs a WMIC command to enable the software distribution agent before the first Install package step runs. You can use the following in your **Run Command Line** task sequence step:

Command Line: **WMIC /namespace:\\root\ccm\policy\machine\requestedconfig path ccm_SoftwareDistributionClientConfig CREATE ComponentName="Enable SWDist", Enabled="true", LockSettings="TRUE", PolicySource="local", PolicyVersion="1.0", SiteSettingsKey="1" /NOINTERACTIVE**

Configure deployment settings

When you use stand-alone media to start the operating system deployment process, you must configure the deployment to make the operating system available to media. You can configure this on the **Deployment Settings** page of the Deploy Software Wizard or the **Deployment Settings** tab in the properties for the deployment. For the **Make available to the following** setting, configure one of the following:

- **Configuration Manager clients, media and PXE**
- **Only media and PXE**
- **Only media and PXE (hidden)**

Create the stand-alone media

You can specify whether the stand-alone media is a USB flash drive or CD/DVD set. The computer that will start the media must support the option that you choose as a bootable drive. For more information, see Create stand-alone media.

Install the operating system from stand-alone media

Insert the stand-alone media in a bootable drive on the computer, and then power it on to install the operating system.

Use multicast to deploy Windows over the network with System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Multicast is a network optimization method that you can use in your System Center Configuration Manager environment where multiple clients are likely to download the same operating system image at the same time. When multicast is used, multiple computers simultaneously download the operating system image as it is multicast by the distribution point, rather than having the distribution point send a copy of the data to each client over a separate connection.

You can deploy operating systems over the network by using multicast in the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)

Complete the steps in one of the operating system deployment scenarios and then use the following sections to support multicast.

Configure a distribution point to support multicast

To use multicast when you deploy operating systems, you must configure a distribution point to support multicast. For more information, see Configure distribution points to support multicast.

Prepare an operating system image for multicast deployments

To configure the operating system image package to support multicast, see Prepare the operating system image for multicast deployments.

Deploy the task sequence

Deploy the operating system to a target collection. For more information, see Deploy a task sequence.

Create an image for an OEM in factory or a local depot with System Center Configuration Manager

11/23/2016 • 2 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Prestaged media deployments in System Center Configuration Manager let you deploy an operating system to a computer that is not fully provisioned. The prestaged media is a Windows Imaging Format (WIM) file that can be installed on a bare-metal computer by the manufacturer (OEM) or at an enterprise staging center that is not connected to the Configuration Manager environment. Later in the Configuration Manager environment, the computer starts by using the boot image provided by the media, a hash check is run on the prestaged media to make sure it is valid, and then the computer connects to the site management point for available task sequences that complete the download process.

This method of deployment can reduce network traffic because the boot image and operating system image are already on the destination computer. You can specify applications, packages, and driver packages to include in the pre-staged media. After the operating system is installed on the computer, the local task sequence cache is checked for applications, packages, or driver packages first, and if the content cannot be found or has been revised, the content is downloaded from a distribution point configured in the prestaged media and then installed.

You can use prestaged media in the following operating system deployment scenarios:

- Install a new version of Windows on a new computer (bare metal)
- Replace an existing computer and transfer settings

Complete the steps in one of the operating system deployment scenarios and then use the following sections to prepare for and create the prestaged media.

Configure deployment settings

When you use prestaged media to start the operating system deployment process, you must configure the deployment to make the operating system available to media. You can configure this on the **Deployment Settings** page of the Deploy Software Wizard or the **Deployment Settings** tab in the properties for the deployment. For the **Make available to the following** setting, configure one of the following:

- **Configuration Manager clients, media and PXE**
- **Only media and PXE**
- **Only media and PXE (hidden)**

Create the prestaged media

Create the prestaged media file to send to the OEM or your local depot. For more information, see Create prestaged media with System Center Configuration Manager.

Send the prestaged media file to the OEM or local depot

Send the media to the OEM or your local depot to prestage the computers. The prestaged media file is applied to a formatted hard disk on the computer.

Start the computer to install the operating system

The prestaged media file is applied to computers. When the computer is started for the first time, the operating system installation process starts.

Deploy Windows to Go with System Center Configuration Manager

11/23/2016 • 26 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

This topic provides the steps to provision Windows To Go in System Center Configuration Manager. Windows To Go is an enterprise feature of Windows 8 that enables the creation of a Windows To Go workspace that can be booted from a USB-connected external drive on computers that meet the Windows 7 or Windows 8 certification requirements, regardless of the operating system running on the computer. Windows To Go workspaces can use the same image enterprises use for their desktops and laptops and can be managed the same way.

For more information about Windows To Go, see Windows To Go feature overview.

Provision Windows To Go

Windows To Go is an operating system stored on a USB-connected external drive. You can provision the Windows To Go drive much like you provision other operating system deployments. However, because Windows To Go is designed to be a user-centric and highly mobile solution, you must take a slightly different approach to provisioning these drives.

At a high level, Windows To Go is a two-phased deployment that allows you to configure the Windows To Go device and prestage content for the operating system deployment. You can achieve this with minimal impact to the user and limit downtime for the user's computer. After you prestage the computer, you must complete the provisioning process to ensure the computer is ready for the user. The provisioning process is similar to the current operating system deployment process. The following lists the general workflow to prestage content and provision Windows To Go:

1. Prerequisites to provision Windows To Go
2. Create prestaged media
3. Create a Windows To Go Creator package
4. Update the task sequence to enable BitLocker for Windows To Go
5. Deploy the Windows To Go Creator package and task sequence
6. User runs the Windows To Go Creator
7. Configuration Manager configures and stages the Windows To Go drive
8. User logs in to Windows 8

Prerequisites to provision Windows To Go

Before you provision Windows To Go, you must complete the following in Configuration Manager:

- **Distribute a boot image to a distribution point**

  Before you create prestaged media, you must distribute the boot image to a distribution point.

> **NOTE**
>
> Boot images are used to install the operating system on the destination computers in your Configuration Manager environment. They contain a version of Windows PE that installs the operating system, as well as any additional device drivers that are required. Configuration Manager provides two boot images: One to support x86 platforms and one to support x64 platforms. You can also create your own boot images. For more information, see Manage boot images.

- **Distribute the Windows 8 operating system image to a distribution point**

  Before you create prestaged media, you must distribute the Windows 8 operating system image to a distribution point.

  > **NOTE**
  >
  > Operating system images are .WIM format files and represent a compressed collection of reference files and folders that are required to successfully install and configure an operating system on a computer. For more information, see Manage operating system images.

- **Create a Task Sequence to Deploy Windows 8**

  You must create a task sequence for a Windows 8 deployment that you will reference when you create prestaged media. For more information, see Manage task sequences to automate tasks.

Create prestaged media

Prestaged media contains the boot image used to start the destination computer and the operating system image that is applied to the destination computer. The computer that you provision with prestaged media can be started by using the boot image. The computer can then run an existing operating system deployment task sequence to install a complete operating system deployment. The task sequence that deploys the operating system is not included in the media.

You can add content, such as applications and device drivers, in addition to the operating system image and boot image during the prestage phase. This reduces the time it takes to deploy an operating system and reduces network traffic because the content is already on the drive.

Use the following procedure to create the prestaged media.

To create prestaged media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, specify the following information, and then click **Next**.
   - Select **Prestaged media**.
   - Select **Allow unattended operating system deployment** to boot to the Windows To Go deployment with no user interaction.

> **IMPORTANT**
> When you use this option with the SMSTSPreferredAdvertID custom variable (set later in this procedure), no user interaction is required and the computer will automatically boot to the Windows To Go deployment when it detects a Windows To Go drive. The user is still prompted for a password if the media is configured for password protection. If you use the **Allow unattended operating system deployment** setting without configuring the SMSTSPreferredAdvertID variable, an error will occur when you deploy the task sequence.

5. On the **Media Management** page, specify the following information, and then click **Next**.

   - Select **Dynamic media** if you want to allow a management point to redirect the media to another management point, based on the client location in the site boundaries.
   - Select **Site-based media** if you want the media to contact only the specified management point.

6. On the **Media Properties** page, specify the following information, and then click **Next**.

   - **Created by**: Specify who created the media.
   - **Version**: Specify the version number of the media.
   - **Comment**: Specify a unique description of what the media is used for.
   - **Media file**: Specify the name and path of the output files. The wizard writes the output files to this location. For example: **\\servername\folder\outputfile.wim**

7. On the **Security** page, specify the following information, and then click **Next**.

   - Select **Enable unknown computer support** to allow the media to deploy an operating system to a computer that is not managed by Configuration Manager. There is no record of these computers in the Configuration Manager database. Unknown computers include the following:
     - A computer where the Configuration Manager client is not installed
     - A computer that is not imported into Configuration Manager
     - A computer that is not discovered by Configuration Manager
   - Select **Protect the media with a password** and enter a strong password to help protect the media from unauthorized access. When you specify a password, the user must provide that password to use the prestaged media.

     > **IMPORTANT**
     > As a security best practice, always assign a password to help protect the prestaged media.

     > **NOTE**
     > When you protect the prestaged media with a password, the user is prompted for the password even when the media is configured with the **Allow unattended operating system deployment** setting.

   - For HTTP communications, select **Create self-signed media certificate**, and then specify the start and expiration date for the certificate.
   - For HTTPS communications, select **Import PKI certificate**, and then specify the certificate to import and its password.

     For more information about this client certificate that is used for boot images, see PKI certificate

requirements.

- **User Device Affinity**: To support user-centric management in Configuration Manager, specify how you want the media to associate users with the destination computer. For more information about how operating system deployment supports user device affinity, see Associate users with a destination computer.
  - Specify **Allow user device affinity with auto-approval** if you want the media to automatically associate users with the destination computer. This functionality is based on the actions of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and destination computer when it deploys the operating system to the destination computer.
  - Specify **Allow user device affinity pending administrator approval** if you want the media to associate users with the destination computer after approval is granted. This functionality is based on the scope of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and the destination computer, but waits for approval from an administrative user before the operating system is deployed.
  - Specify **Do not allow user device affinity** if you do not want the media to associate users with the destination computer. In this scenario, the task sequence does not associate users with the destination computer when it deploys the operating system.

8. On the **Task Sequence** page, specify the Windows 8 task sequence that you created in the previous section.
9. On the **Boot image** page, specify the following information, and then click **Next**.

> **IMPORTANT**
> The architecture of the boot image that is distributed must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image. For Windows 8 certified computers in EFI mode, you must use an x64 boot image.

- **Boot image**: Specify the boot image to start the destination computer.
- **Distribution point**: Specify the distribution point that hosts the boot image. The wizard retrieves the boot image from the distribution point and writes it to the media.

> **NOTE**
> The administrative user must have **Read** access rights to the boot image content on the distribution point. For more information, see Manage accounts to access content.

- If you selected **Site-based media** on the **Media Management** page of this wizard, in the **Management point** box, specify a management point from a primary site.
- If you selected **Dynamic media** on the **Media Management** page of the wizard, in the **Associated management points** box, specify the primary site management points to use and a priority order for the initial communications.

10. On the **Images** page, specify the following information, and then click **Next**.

- **Image package**: Specify the package that contains the Windows 8 operating system image.
- **Image index**: Specify the image to deploy if the package contains multiple operating system images.

- **Distribution point**: Specify the distribution point that hosts the operating system image package. The wizard retrieves the operating system image from the distribution point and writes it to the media.

  > **NOTE**
  > The administrative user must have **Read** access rights to the operating system image content on the distribution point. For more information, see Manage accounts to access content.

11. On the **Select Application** page, select application content to include in the media file, and then click **Next**.
12. On the **Select Package** page, select additional package content to include in the media file, and then click **Next**.
13. On the **Select Driver Package** page, select driver package content to include in the media file, and then click **Next**.
14. On the **Distribution Points** page, select one or more distribution points that contain the content required by the task sequence, and then click **Next**.
15. On the **Customization** page, specify the following information, and then click **Next**.

    - **Variables**: Specify the variables that the task sequence uses to deploy the operating system. For Windows To Go, use the SMSTSPreferredAdvertID variable to automatically select the Windows To Go deployment by using the following format:

      SMSTSPreferredAdvertID = {*DeploymentID*}, where DeploymentID is the deployment ID associated with the task sequence that you will use to complete the provisioning process for the Windows To Go drive.

      > **TIP**
      > When you use this variable with a task sequence that is set to run unattended (set earlier in this procedure), no user interaction is required and the computer automatically boots to the Windows To Go deployment when it detects a Windows To Go drive. The user is still prompted for a password if the media is configured for password protection.

    - **Prestart commands**: Specify any prestart commands that you want to run before the task sequence runs. Prestart commands can be a script or executable that can interact with the user in Windows PE before the task sequence runs to install the operating system. Configure the following for the Windows To Go deployment:

      - **OSDBitLockerPIN**: BitLocker for Windows To Go requires a passphrase. Set the **OSDBitLockerPIN** variable as part of a prestart command to set the BitLocker passphrase for the Windows To Go drive.

        > **WARNING**
        > After BitLocker is enabled for the passphrase, the user must enter the passphrase each time the computer boots to the Windows To Go drive.

      - **SMSTSUDAUsers**: Specifies the primary user of the destination computer. Use this variable to collect the user name, which can then be used to associate the user and device. For more information, see Associate users with a destination computer.

> **TIP**
> To retrieve the username, you can create an input box as part of the prestart command, have the user enter their username, and then set the variable with the value. For example, you can add the following lines to the prestart command script file:
>
> ```
> UserID = inputbox("Enter Username" ,"Enter your username:","",400,0)
> ```
>
> ```
> env("SMSTSUDAUsers") = UserID
> ```

For more information about how to create a script file to use as your prestart command, see Prestart commands for task sequence media.

16. Complete the wizard.

> **NOTE**
> It can take an extended period of time for the wizard to complete the prestaged media file.

Create a Windows To Go Creator package

As part of the Windows To Go deployment, you must create a package to deploy the prestage media file. The package must include the tool that configures the Windows To Go drive and extracts the prestaged media to the drive. Use the following procedure to create the Windows To Go Creator package.

To create the Windows To Go Creator package

1. On the server to host the Windows To Go Creator package files, create a source folder for the package source files.

   > **NOTE**
   > The computer account of the site server must have **Read** access rights to the source folder.

2. Copy the prestaged media file that you created in the Create prestaged media section to the package source folder.
3. Copy the Windows To Go Creator tool (WTGCreator.exe) to the package source folder. The creator tool is available on any primary site server at the following location: *<ConfigMgrInstallationFolder>*\OSD\Tools\WTG\Creator.
4. Create a package and program by using the Create Package and Program Wizard.
5. In the Configuration Manager console, click **Software Library**.
6. In the **Software Library** workspace, expand **Application Management**, and then click **Packages**.
7. On the **Home** tab, in the **Create** group, click **Create Package**.
8. On the **Package** page, specify the name and description of the package. For example, enter **Windows To Go** for the package name and specify **Package to configure a Windows To Go drive using System Center Configuration Manager** for the package description.
9. Select **This package contains source files**, specify the path to the package source folder that you created in step 1, and then click **Next**.
10. On the **Program Type** page, select **Standard program**, and then click **Next**.
11. On the **Standard Program** page, specify the following:

- **Name**: Specify the name of the program. For example, type **Creator** for the program name.
- **Command Line**: Type **WTGCreator.exe /wim:PrestageName.wim**, where PrestageName is the name of prestaged file that you created and copied to the package source folder for the Windows To Go Creator package.

  Optionally, you can add the following options:

  - **enableBootRedirect**: command-line option to change the Windows To Go startup options to allow boot redirection. When you use this option, the computer will boot from USB without having to change the boot order in the computer firmware or have the user select from a list of boot options during startup. If a Windows To Go drive is detected, the computer boots to that drive.
- **Run**: Specify **Normal** to run the program based on the system and program defaults.
- **Program can run**: Specify whether the program can run only when a user is logged on.
- **Run mode**: Specify whether the program will run with the logged on users permissions or with administrative permissions. The Windows To Go Creator requires elevated permissions to run.
- Select **Allow users to view and interact with the program installation**, and then click **Next**.

12. On the Requirements page, specify the following:

    - **Platform requirements**: Select the applicable Windows 8 platforms to allow provisioning.
    - **Estimated disk space**: Specify the size of the package source folder for the Windows To Go Creator.
    - **Maximum allowed run time (minutes)**: Specifies the maximum time that the program is expected to run on the client computer. By default, this value is set to 120 minutes.

      > **IMPORTANT**
      > If you are using maintenance windows for the collection on which this program is run, a conflict might occur if the **Maximum allowed run time** is longer than the scheduled maintenance window. If the maximum run time is set to **Unknown**, it will start during the maintenance window, but will continue to run until it completes or fails after the maintenance window is closed. If you set the maximum run time to a specific period (not set to Unknown) that exceeds the length of any available maintenance window, then that program will not be run.

      > **NOTE**
      > If the value is set to **Unknown**, Configuration Manager sets the maximum allowed run time to 12 hours (720 minutes).

      > **NOTE**
      > If the maximum run time (whether set by the user or as the default value) is exceeded, Configuration Manager stops the program if **run with administrative rights** is selected and **Allow users to view and interact with the program installation** is not selected on the **Standard Program** page.

    Click **Next** and complete the wizard.

Update the task sequence to enable BitLocker for Windows To Go

Windows To Go enables BitLocker on an external bootable drive without the use of TPM. Therefore, you must use a separate tool to configure BitLocker on the Windows To Go drive. To enable BitLocker, you must add an action to

the task sequence after the **Setup Windows and ConfigMgr** step.

> **NOTE**
> BitLocker for Windows To Go requires a passphrase. In the Create prestaged media step, you set the passphrase as part of a prestart command by using the OSDBitLockerPIN variable.

Use the following procedure to update the Windows 8 task sequence to enable BitLocker for Windows To Go.

To update the Windows 8 task sequence to enable BitLocker

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Application Management**, and then click **Packages**.
3. On the **Home** tab, in the **Create** group, click **Create Package**.
4. On the **Package** page, specify the name and description of the package. For example, type **BitLocker for Windows To Go** for the package name and specify **Package to update BitLocker for Windows To Go** for the package description.
5. Select **This package contains source files**, specify the location for the BitLocker tool for Windows To Go, and then click **Next**. The BitLocker tool is available on any Configuration Manager primary site server at the following location: <*ConfigMgrInstallationFolder*>\OSD\Tools\WTG\BitLocker\
6. On the **Program Type** page, select **Do not create a program**.
7. Click **Next** and complete the wizard.
8. In the Configuration Manager console, click **Software Library**.
9. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
10. Select the Windows 8 task sequence that you reference in the prestaged media.
11. On the **Home** tab, in the **Task Sequence** group, click **Edit**.
12. Click the **Setup Windows and ConfigMgr** step, click **Add**, click **General**, and then click **Run Command Line**. The Run Command Line step is added after the Setup Windows and ConfigMgr step.
13. On the **Properties** tab for the **Run Command Line** step, add the following:
    a. **Name**: Specify a name for the command line, such as **Enable BitLocker for Windows To Go**.
    b. **Command Line**: i386\osdbitlocker_wtg.exe /Enable /pwd:< *None*|*AD*>

       Parameters:

       - /pwd: - Specify the BitLocker password recovery mode. This parameter is required you use the /Enable parameter is in the command-line.

         Select **AD** to configure BitLocker Drive Encryption to back up recovery information for BitLocker-protected drives to Active Directory Domain Services (AD DS). Backing up recovery passwords for a BitLocker-protected drive allows administrative users to recover the drive if it is locked. This ensures that encrypted data belonging to the enterprise can always be accessed by authorized users. When you specify **None**, the user is responsible for keeping a copy of the recovery password or recovery key. If the user loses that information or neglects to decrypt the drive before leaving the organization, administrative users cannot easily access to the drive.

       - /wait: - Specify whether the task sequence waits for encryption to complete before it

completes.

c. Select **Package**, and then specify the package that you created at the start of this procedure.

d. On the **Options** tab, add the following conditions:

- Condition = Task Sequence Variable
- Variable = _SMSTSWTG
- Condition = Equals
- Value = True

> **NOTE**
>
> The **Enable BitLocker** step, which is likely after the new command-line step, is not used to enable BitLocker for Windows To Go. However, you can keep this step in the task sequence to use for Windows 8 deployments that do not use a Windows To Go drive.

Deploy the Windows To Go Creator package and task sequence

Windows To Go is a hybrid deployment process. Therefore, you must deploy the Windows To Go Creator package and the Windows 8 task sequence. Use the following procedures to complete the deployment process.

To deploy the Windows To Go Creator package

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Application Management**, and then click **Packages**.
3. Select the Windows To Go package that you created in the Create a Windows To Go Creator package step.
4. On the **Home** tab, in the **Deployment** group, click **Deploy**.
5. On the **General** page, specify the following settings:
   a. **Software**: Verify that the Windows To Go package is selected.
   b. **Collection**: Click **Browse** to select the collection to which you want to deploy the Windows To Go package.
   c. **Use default distribution point groups associated to this collection**: Select this option if you want to store the package content on the collections default distribution point group. If you have not associated the selected collection with a distribution point group, this option will be unavailable.
6. On the **Content** page, click **Add** and then select the distribution points or distribution point groups to which you want to deploy the content associated with this package and program.
7. On the **Deployment Settings** page, select **Available** for the deployment type, and then click **Next**.
8. On the **Scheduling**, configure when this package and program will be deployed or made available to client devices.

   The options on this page will differ depending on whether the deployment action is set to **Available** or **Required**.
9. On the **Scheduling**, configure the following settings, and then click **Next**.
   a. **Schedule when this deployment will become available**: Specify the date and time when the package and program is available to run on the destination computer. When you select **UTC**, this setting ensures that the package and program is available for multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

b. **Schedule when this deployment will expire**: Specify the date and time when the package and program expires on the destination computer. When you select **UTC**, this setting ensures that the task sequence expires on multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

10. On the **User Experience** page of the Wizard, specify the following information:

    - **Software installation**: Allows the software to be installed outside of any configured maintenance windows.
    - **System restart (if required to complete the installation)**: Allows a device to restart outside of configured maintenance windows when required by the software installation.
    - **Embedded Devices**: When you deploy packages and programs to Windows Embedded devices that are write filter enabled, you can specify to install the packages and programs on the temporary overlay and commit changes later, or commit the changes at the installation deadline or during a maintenance window. When you commit changes at the installation deadline or during a maintenance window, a restart is required and the changes persist on the device.

11. On the **Distribution Points** page, specify the following information:

    - **Deployment options:** Specify **Download content from distribution point and run locally**.
    - **Allow clients to share content with other clients on the same subnet**: Select this option to reduce load on the network by allowing clients to download content from other clients on the network that have already downloaded and cached the content. This option utilizes Windows BranchCache and can be used on computers running Windows Vista SP2 and later.
    - **All clients to use a fallback source location for content**: Specify whether to allow clients to fall back and use a non-preferred distribution point as the source location for content when the content is not available on a preferred distribution point.

12. Complete the wizard.

To deploy the Windows 8 task sequence

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. Select the Windows 8 task sequence that you created in the Prerequisites to provision Windows To Go step.
4. On the **Home** tab, in the **Deployment** group, click **Deploy**.
5. On the **General** page, specify the following settings:

    a. **Task sequence**: Verify that the Windows 8 task sequence is selected.

    b. **Collection**: Click **Browse** to select the collection that includes all devices for which a user might provision Windows To Go.

    > **IMPORTANT**
    > If the prestaged media that you created in the Create prestaged media section uses the SMSTSPreferredAdvertID variable, you can deploy the task sequence to the **All Systems** collection and specify the **Windows PE only (hidden)** setting on the **Content** page. Because the task sequence is hidden, it will only be available to media.

    c. **Use default distribution point groups associated to this collection**: Select this option if you want to store the package content on the collections default distribution point group. If you have not

associated the selected collection with a distribution point group, this option will be unavailable.

6. On the **Deployment Settings** page, configured the following settings, and then click **Next**.

   - **Purpose**: Select **Available**. When you deploy the task sequence to a user, the user sees the published task sequence in the Application Catalog and can request it on demand. If you deploy the task sequence to a device, the user will see the task sequence in Software Center and can install it on demand.

   - **Make available to the following**: Specify whether the task sequence is available to Configuration Manager clients, media, or PXE.

     > **IMPORTANT**
     > Use the **Only media and PXE (hidden)** setting for automated task sequence deployments. Select **Allow unattended operating system deployment** and set the SMSTSPreferredAdvertID variable as part of the prestaged media to have the computer automatically boot to the Windows To Go deployment with no user interaction when it detects a Windows To Go drive. For more information about these prestaged media settings, see the Create prestaged media section.

7. On the **Scheduling** page, configure the following settings, and then click **Next**.

   a. **Schedule when this deployment will become available**: Specify the date and time when the task sequence is available to run on the destination computer. When you select **UTC**, this setting ensures that the task sequence is available for multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

   b. **Schedule when this deployment will expire**: Specify the date and time when the task sequence expires on the destination computer. When you select **UTC**, this setting ensures that the task sequence expires on multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

8. On the **User Experience** page, specify the following information:

   - **Show Task Sequence progress**: Specify whether the Configuration Manager client displays the progress of the task sequence.

   - **Software installation**: Specify whether the user is allowed to install software outside a configured maintenance windows after the scheduled time.

   - **System restart (if required to complete the installation)**: Allows a device to restart outside of configured maintenance windows when required by the software installation.

   - **Embedded Devices**: When you deploy packages and programs to Windows Embedded devices that are write filter enabled, you can specify to install the packages and programs on the temporary overlay and commit changes later, or commit the changes at the installation deadline or during a maintenance window. When you commit changes at the installation deadline or during a maintenance window, a restart is required and the changes persist on the device.

   - **Internet-based clients**: Specify whether the task sequence is allowed to run on an Internet-based client. Operations that install software, such as an operating system, are not supported with this setting. Use this option only for generic script-based task sequences that perform operations in the standard operating system.

9. On the **Alerts** page, specify the alert settings that you want for this task sequence deployment, and then click **Next**.

10. On the **Distribution Points** page, specify the following information, and then click **Next**.

- **Deployment options**: Select **Download content locally when needed by running task sequence**.
- **When no local distribution point is available, use a remote distribution point**: Specify whether clients can use distribution points that are on slow and unreliable networks to download the content that is required by the task sequence.
- **Allow clients to use a fallback source location for content**:
  - *Prior to version 1610*, you can select the Allow fallback source location for content check box to allow clients outside these boundary groups to fall back and use the distribution point as a source location for content when no other distribution points are available.
  - *Beginning with version 1610*, you no longer can configure **Allow fallback source location for content**. Instead, you configure relationships between boundary groups that determine when a client can begin to search additional boundary groups for a valid content source location.

11. Complete the wizard.

User runs the Windows To Go Creator

After you deploy the Windows To Go package and Windows 8 task sequence, the Windows To Go Creator is available to the user. The user can go to the software catalog, or Software Center if the Windows To Go Creator was deployed to devices, and run the Windows To Go Creator program. Once the creator package is downloaded, a flashing icon is displayed on the task bar. When the user clicks the icon, a dialog box is displayed for the user to select the Windows To Go drive to provision (unless the /drive command-line option is used). If the drive does not meet the requirements for Windows To Go or if the drive does not have enough free disk space to install the image, the creator program displays an error message. The user can verify the drive and image that will be applied from the confirmation page. As the creator configures and prestages content to the Windows To Go drive, it displays a progress dialog box. After the prestaging is complete, the creator displays a prompt to restart the computer to boot to the Windows To Go drive.

> **NOTE**
> If you did not enable boot redirection as part of the command line for the creator program in the Create a Windows To Go Creator package section, the user might be required to manually boot to the Windows To Go drive on every system restart.

Configuration Manager configures and stages the Windows To Go drive

After the computer restarts to the Windows To Go drive, the drive will boot into Windows PE and connect to the management point to get the policy to complete the operating system deployment. Configuration Manager configures and stages the drive. After Configuration Manager stages the drive, the user can restart the computer to finalize the provisioning process (such as to join a domain or install apps). This process is the same for any prestaged media.

User logs in to Windows 8

After Configuration Manager completes the provisioning process and the Windows 8 lock screen is displayed, the user can login to the operating system.

Manage Windows as a service using System Center Configuration Manager

3/26/2017 • 20 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

In System Center Configuration Manager, you can view the state of Windows as a Service in your environment, create servicing plans to form deployment rings and ensure that Windows 10 current branch systems are kept up to date when new builds are released, and view alerts when Windows 10 clients are near end of support for their build of Current Branch (CB) or Current Branch for Business (CBB).

For more information about Windows 10 servicing options, see Windows 10 servicing options for updates and upgrades.

Use the following sections to manage Windows as a service.

Prerequisites

To see data in the Windows 10 servicing dashboard, you must do the following:

- Windows 10 computers must use Configuration Manager software updates with Windows Server Update Services (WSUS) for software update management. When computers use Windows Update for Business (or Windows Insiders) for software update management, the computer will not be evaluated in Windows 10 servicing plans. For more information, see Integration with Windows Update for Business in Windows 10.
- WSUS 4.0 with the hotfix 3095113 must be installed on your software update points and site servers. This adds the **Upgrades** software update classification. For more information, see Prerequisites for software updates.
- WSUS 4.0 with the hotfix 3159706 must be installed on your software update points and site servers to upgrade computers to the Windows 10 Anniversary Update, as well as for subsequence versions. There are manual steps described in the support article that you must take to install this hotfix. For more information, see the Enterprise Mobility and Security Blog.
- Enable Heartbeat Discovery. The data displayed in the Windows 10 servicing dashboard is found by using discovery. For more information, see Configure Heartbeat Discovery.

  The following Windows 10 branch and build information is discovered and stored in the following attributes:

  - **Operating System Readiness Branch**: Specifies the operating system branch. For example, **0** = CB (no not defer upgrades), **1** = CBB (defer upgrades), **2** = Long Term Servicing Branch (LTSB)
  - **Operating System Build**: Specified the operating system build. For example, **10.0.10240** (RTM) or **10.0.10586** (version 1511)
- The service connection point must be installed and configured for **Online, persistent connection** mode to see data on the Windows 10 servicing dashboard. When you are in offline mode, you will not see data updates in the dashboard until you get Configuration Manager servicing updates.
  For more information, see About the service connection point.
- Specify the group policy setting, **Defer Upgrades and Updates**, to determine whether a computer is CB or CBB.

- Internet Explorer 9 or later must be installed on the computer that runs the Configuration Manager console.
- Software updates must be configured and synchronized. You must select the **Upgrades** classification and synchronize software updates before any Windows 10 feature upgrades will be available in the Configuration Manager console. For more information, see Prepare for software updates management.

Windows 10 servicing dashboard

The Windows 10 servicing dashboard provides you with information about Windows 10 computers in your environment, active servicing plans, compliance information, and so on. The data in the Windows 10 servicing dashboard is dependent on having the Service Connection Point installed. The dashboard has the following tiles:

- **Windows 10 Usage tile**: Provides a breakdown of public builds of Windows 10. Windows Insiders builds are listed as **other** as well as any builds that are not yet known to your site. The service connection point will download metadata that informs it about the Windows builds, and then this data is compared against discovery data.
- **Windows 10 Rings tile**: Provides a breakdown of Windows 10 by branch and readiness state . The LTSB segment will be all LTSB versions (whereas the first tile breaks down the specific versions. For example, Windows 10 LTSB 2015. The **Release Ready** segment corresponds to CB, and the **Business ready** segment is CBB.
- **Create Service Plan tile**: Provides a quick way to create a servicing plan. You specify the name, collection (only displays the top ten collections by size, smallest first), deployment package (only displays the top ten packages by most recently modified), and readiness state. Default values are used for the other settings. Click **Advanced Settings** to start the Create Servicing Plan wizard where you can configure all of the service plan settings.
- **Expired tile**: Displays the percentage of devices that are on a build of Windows 10 that is past its end of life. Configuration Manager determines the percentage from the metadata that the Service Connection Point downloads and compares it against discovery data. A build that is past its end of life is no longer receiving monthly cumulative updates, which includes security updates. The computers in this category should be upgraded to the next build version. Configuration Manager rounds up to the next whole number. For example, if you have 10,000 computers and only one on an expired build, the tile will display 1%.
- **Expire Soon tile**: Displays the percentage of computers that are on a build that is near end of life (within about four months), similar to the **Expired** tile. Configuration Manager rounds up to the next whole number.
- **Alerts tile**: Displays active alerts.
- **Service Plan Monitoring tile**: Display servicing plans that you have created and a chart of the compliance for each. This gives you a quick overview of the current state of the servicing plan deployments. If an earlier deployment ring meets your expectations for compliance, then you can select a later servicing plan (deploying ring) and click **Deploy Now** instead of waiting for the servicing plan rules to be triggered automatically.
- The **Windows 10 Builds tile**: Display is a fixed image time line that provides you an overview of the Windows 10 builds that are currently released and gives you a general idea of when builds will transition into different states.

> **IMPORTANT**
> The information shown in the Windows 10 servicing dashboard (such as the support lifecycle for Windows 10 versions) is provided for your convenience and only for use internally within your company. You should not solely rely on this information to confirm update compliance. Be sure to verify the accuracy of the information provided to you.

Servicing plan workflow

Windows 10 servicing plans in Configuration Manager are much like automatic deployment rules for software updates. You create a servicing plan with the following criteria that Configuration Manager evaluates:

- **Upgrades classification**: Only updates that are in the **Upgrades** classification are evaluated.
- **Readiness state**: The readiness state defined in the servicing plan is compared with the readiness state for the upgrade. The metadata for the upgrade is retrieved when the service connection point checks for updates.
- **Time deferral**: The number of days that you specify for **How many days after Microsoft has published a new upgrade would you like to wait before deploying in your environment** in the servicing plan. Configuration Manager evaluates whether to include an upgrade in the deployment if the current date is after the release date plus the configured number of days.

When an upgrade meets the criteria, the servicing plan adds the upgrade to the deployment package, distributes the package to distribution points, and deploys the upgrade to the collection based on the settings that you configure in the servicing plan. You can monitor the deployments in the Service Plan Monitoring tile on the Windows 10 Servicing Dashboard. For more information, see Monitor software updates.

Windows 10 servicing plan

As you deploy Windows 10 CB, you can create one or more servicing plans to define the deployment rings that you want in your environment, and then monitor them in the Windows 10 servicing dashboard.
Servicing plans use only the **Upgrades** software updates classification, not cumulative updates for Windows 10. For those updates, you will still need to deploy by using the software updates workflow. The end-user experience with a servicing plan is the same as it is with software updates, including the settings that you configure in the servicing plan.

> **NOTE**
> You can use a task sequence to deploy an upgrade for each Windows 10 build, but it requires more manual work. You would need to import the updated source files as an operating system upgrade package, and then create and deploy the task sequence to the appropriate set of computers. However, a task sequence provides additional customized options, such as the pre-deployment and post-deployment actions.

You can create a basic servicing plan from the Windows 10 servicing dashboard. After you specify the name, collection (only displays the top ten collections by size, smallest first), deployment package (only displays the top ten packages by most recently modified), and readiness state, Configuration Manager creates the servicing plan with default values for the other settings. You can also start the Create Servicing Plan wizard to configure all of the settings. Use the following procedure to create a servicing plan by using the Create Servicing Plan wizard.

> **NOTE**
> Beginning in Configuration Manager version 1602, you can manage the behavior for high-risk deployments. A high-risk deployment is a deployment that is automatically installed and has the potential to cause unwanted results. For example, a task sequence that has a purpose of **Required** that deploys Windows 10 is considered a high-risk deployment. For more information, see Settings to manage high-risk deployments.

To create a Windows 10 servicing plan

1. In the Configuration Manager console, click **Software Library**.
2. In the Software Library workspace, expand **Windows 10 Servicing**, and then click **Servicing Plans**.

3. On the **Home** tab, in the **Create** group, click **Create Servicing Plan**. The Create Servicing Plan Wizard opens.

4. On the **General** page, configure the following settings:

   - **Name**: Specify the name for the servicing plan. The name must be unique, help to describe the objective of the rule, and identify it from others in the Configuration Manager site.

   - **Description**: Specify a description for the servicing plan. The description should provide an overview of the servicing plan and any other relevant information that helps to identify and differentiate the plan among others in the Configuration Manager site. The description field is optional, has a limit of 256 characters, and has a blank value by default.

5. On the Servicing Plan page, configure the following settings:

   - **Target Collection**: Specifies the target collection to be used for the servicing plan. Members of the collection receive the Windows 10 upgrades that are defined in the servicing plan.

   > **NOTE**
   > Beginning in Configuration Manager version 1602, when you deploy a high-risk deployment, such as servicing plan, the **Select Collection** window displays only the custom collections that meet the deployment verification settings that are configured in the site's properties. High-risk deployments are always limited to custom collections, collections that you create, and the built-in **Unknown Computers** collection. When you create a high-risk deployment, you cannot select a built-in collection such as **All Systems**. Uncheck **Hide collections with a member count greater than the site's minimum size configuration** to see all custom collections that contain fewer clients than the configured maximum size. For more information, see Settings to manage high-risk deployments.
   > The deployment verification settings are based on the current membership of the collection. After you deploy the servicing plan, the collection membership is not reevaluated for the high-risk deployment settings.
   > For example, let's say you set **Default size** to 100 and the **Maximum size** to 1000. When you create a high risk deployment, the **Select Collection** window will only display collections that contain less than 100 clients. If you clear the **Hide collections with a member count greater than the site's minimum size configuration** setting, the window will display collections that contain less than 1000 clients.
   > When you select a collection that contains a site role, the following applies:
   >
   > - If the collection contains a site system server and in the deployment verification settings you configure to block collections with site system servers, then an error occurs and you cannot continue.
   >   - If the collection contains a site system server and in the deployment verification settings you configure to warn you if collections that have site system servers, if the collection exceeds the default size value, or if the collection contains a server, then the Deploy Software Wizard will display a high risk warning. You must agree to create a high risk deployment and an audit status message is created.

6. On the Deployment Ring page, configure the following settings:

   - **Specify the Windows readiness state to which this servicing plan should apply**: Select one of the following:

     - **Release Ready (Current Branch)**: In the CB servicing model, feature updates are available as soon as Microsoft releases them.

     - **Business Ready (Current Branch for Business)**: The CBB servicing branch is typically used for broad deployment. Windows 10 clients in the CBB servicing branch receive the same build of Windows 10 as those in the CB servicing branch, just at a later time.

     For more information about servicing branches and what options is best for you, see Servicing branches.

   - **How many days after Microsoft has published a new upgrade would you like to wait before**

deploying in your environment: Configuration Manager evaluates whether to include an upgrade in the deployment if the current date is after the release date plus the number of days that you configure for this setting.

- Prior to Configuration Manager version 1602, click **Preview** to view the Windows 10 updates associated with the readiness state.

For more information, see Servicing branches.

7. Beginning in Configuration Manager version 1602, on the Upgrades page, configure the search criteria to filter the upgrades that will be added to the service plan. Only upgrades that meet the specified criteria will be added to the associated deployment.

   Click **Preview** to view the upgrades that meet the specified criteria.

8. On the Deployment Schedule page, configure the following settings:

   - **Schedule evaluation**: Specify whether Configuration Manager evaluates the available time and installation deadline times by using UTC or the local time of the computer that runs the Configuration Manager console.

     > **NOTE**
     > When you select local time, and then select **As soon as possible** for the **Software available time** or **Installation deadline**, the current time on the computer running the Configuration Manager console is used to evaluate when updates are available or when they are installed on a client. If the client is in a different time zone, these actions will occur when the client's time reaches the evaluation time.

   - **Software available time**: Select one of the following settings to specify when the software updates are available to clients:

     - **As soon as possible**: Select this setting to make the software updates that are included in the deployment available to the client computers as soon as possible. When you create the deployment with this setting selected, Configuration Manager updates the client policy. Then, at the next client policy polling cycle, clients become aware of the deployment and can obtain the updates that are available for installation.

     - **Specific time**: Select this setting to make the software updates that are included in the deployment available to the client computers at a specific date and time. When you create the deployment with this setting enabled, Configuration Manager updates the client policy. Then, at the next client policy polling cycle, clients become aware of the deployment. However, the software updates in the deployment are not available for installation until after the configured date and time.

   - **Installation deadline**: Select one of the following settings to specify the installation deadline for the software updates in the deployment:

     - **As soon as possible**: Select this setting to automatically install the software updates in the deployment as soon as possible.

     - **Specific time**: Select this setting to automatically install the software updates in the deployment at a specific date and time. Configuration Manager determines the deadline to install software updates by adding the configured **Specific time** interval to the **Software available time**.

> **NOTE**
>
> The actual installation deadline time is the displayed deadline time plus a random amount of time up to 2 hours. This reduces the potential impact of all client computers in the destination collection installing the updates in the deployment at the same time.
>
> You can configure the **Computer Agent** client setting **Disable deadline randomization** to disable the installation randomization delay for required updates. For more information, see Computer Agent.

9. On the User Experience page, configure the following settings:

   - **User notifications**: Specify whether to display notification of the updates in Software Center on the client computer at the configured **Software available time** and whether to display user notifications on the client computers.
   - **Deadline behavior**: Specify the behavior that is to occur when the deadline is reached for the update deployment. Specify whether to install the updates in the deployment. Also specify whether to perform a system restart after update installation regardless of a configured maintenance window. For more information about maintenance windows, see How to use maintenance windows.
   - **Device restart behavior**: Specify whether to suppress a system restart on servers and workstations after updates are installed and a system restart is required to complete the installation.
   - **Write filter handling for Windows Embedded devices**: When you deploy updates to Windows Embedded devices that are write filter enabled, you can specify to install the update on the temporary overlay and either commit changes later or commit the changes at the installation deadline or during a maintenance window. When you commit changes at the installation deadline or during a maintenance window, a restart is required and the changes persist on the device.

   > **NOTE**
   >
   > When you deploy an update to a Windows Embedded device, make sure that the device is a member of a collection that has a configured maintenance window.

10. On the Deployment Package page, select an existing deployment package or configure the following settings to create a new deployment package:

    a. **Name**: Specify the name of the deployment package. This must be a unique name that describes the package content. It is limited to 50 characters.

    b. **Description**: Specify a description that provides information about the deployment package. The description is limited to 127 characters.

    c. **Package source**: Specifies the location of the software update source files. Type a network path for the source location, for example, **\\server\sharename\path**, or click **Browse** to find the network location. You must create the shared folder for the deployment package source files before you proceed to the next page.

    > **NOTE**
    >
    > The deployment package source location that you specify cannot be used by another software deployment package.

> **IMPORTANT**
> The SMS Provider computer account and the user that is running the wizard to download the software updates must both have **Write** NTFS permissions on the download location. You should carefully restrict access to the download location in order to reduce the risk of attackers tampering with the software update source files.

> **IMPORTANT**
> You can change the package source location in the deployment package properties after Configuration Manager creates the deployment package. But if you do so, you must first copy the content from the original package source to the new package source location.

d. **Sending priority**: Specify the sending priority for the deployment package. Configuration Manager uses the sending priority for the deployment package when it sends the package to distribution points. Deployment packages are sent in priority order: High, Medium, or Low. Packages with identical priorities are sent in the order in which they were created. If there is no backlog, the package will process immediately regardless of its priority.

11. On the Distribution Points page, specify the distribution points or distribution point groups that will host the update files. For more information about distribution points, see Configure a distribution point.

> **NOTE**
> This page is available only when you create a new software update deployment package.

12. On the Download Location page, specify whether to download the update files from the Internet or from your local network. Configure the following settings:

    - **Download software updates from the Internet**: Select this setting to download the updates from a specified location on the Internet. This setting is enabled by default.
    - **Download software updates from a location on the local network**: Select this setting to download the updates from a local directory or shared folder. This setting is useful when the computer that runs the wizard does not have Internet access. Any computer with Internet access can preliminarily download the updates and store them in a location on the local network that is accessible from the computer that runs the wizard.

13. On the Language Selection page, select the languages for which the selected updates are downloaded. The updates are downloaded only if they are available in the selected languages. Updates that are not language specific are always downloaded. By default, the wizard selects the languages that you have configured in the software update point properties. At least one language must be selected before proceeding to the next page. When you select only languages that are not supported by an update, the download will fail for the update.

14. On the Summary page, review the settings and click **Next** to create the servicing plan.

After you have completed the wizard, the servicing plan will run. It will add the updates that meet the specified criteria to a software update group, download the updates to the content library on the site server, distribute the updates to the configured distribution points, and then deploy the software update group to clients in the target collection.

Modify a servicing plan

After you create a basic servicing plan from the Windows 10 servicing dashboard or you need to change the settings for an existing servicing plan, you can go to properties for the servicing plan.

> **NOTE**
> You can configure settings in the properties for the servicing plan that are not available in the wizard when you create the servicing plan. The wizard uses default settings for the settings for the following: download settings, deployment settings, and alerts.

Use the following procedure to modify the properties of a servicing plan.

To modify the properties of a servicing plan

1. In the Configuration Manager console, click **Software Library**.
2. In the Software Library workspace, expand **Windows 10 Servicing**, click **Servicing Plans**, and then select the servicing plan that you want to modify.
3. On the **Home** tab, click **Properties** to open properties for the selected servicing plan.

   The following settings are available in the servicing plan properties that were not configured in the wizard:

   Deployment Settings: On the Deployment Settings tab, configure the following settings:

   - **Type of deployment**: Specify the deployment type for the software update deployment. Select **Required** to create a mandatory software update deployment in which the software updates are automatically installed on clients before a configured installation deadline. Select **Available** to create an optional software update deployment that is available for users to install from Software Center.

     > **IMPORTANT**
     > After you create the software update deployment, you cannot later change the type of deployment.

     > **NOTE**
     > A software update group deployed as **Required** will be downloaded in background and honor BITS settings, if configured.
     > However, software update groups deployed as **Available** will be downloaded in the foreground and will ignore BITS settings.

   - **Use Wake-on-LAN to wake up clients for required deployments**: Specify whether to enable Wake On LAN at the deadline to send wake-up packets to computers that require one or more software updates in the deployment. Any computers that are in sleep mode at the installation deadline time will be awakened so the software update installation can initiate. Clients that are in sleep mode that do not require any software updates in the deployment are not started. By default, this setting is not enabled and is available only when **Type of deployment** is set to **Required**.

     > **WARNING**
     > Before you can use this option, computers and networks must be configured for Wake On LAN.

   - **Detail level**: Specify the level of detail for the state messages that are reported by client computers.

Download Settings: On the Download Settings tab, configure the following settings:

-   Specify whether the client will download and install the software updates when a client is connected to a slow network or is using a fallback content location.

-   Specify whether to have the client download and install the software updates from a fallback distribution point when the content for the software updates is not available on a preferred distribution point.

-   **Allow clients to share content with other clients on the same subnet**: Specify whether to enable the use of BranchCache for content downloads. For more information about BranchCache, see [Fundamental concepts for content management](../../core/plan-design/hierarchy/fundamental-concepts-for-content-management.md#branchcache).

-   Specify whether to have clients download software updates from Microsoft Update if software updates are not available on distribution points.
> [!IMPORTANT]
> Do not use this setting for Windows 10 Servicing updates. Configuration Manager (at least through version 1610) will fail to download the Windows 10 Servicing updates from Microsoft Update.

-   Specify whether to allow clients to download after an installation deadline when they use metered Internet connections. Internet providers sometimes charge by the amount of data that you send and receive when you are on a metered Internet connection.

Alerts: On the Alerts tab, configure how Configuration Manager and System Center Operations Manager will generate alerts for this deployment. You can configure alerts only when **Type of deployment** is set to **Required** on the Deployment Settings page.

> [!NOTE]
> You can review recent software updates alerts from the **Software Updates** node in the **Software Library** workspace.

Monitor operating system deployments in System Center Configuration Manager

11/23/2016 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

The Configuration Manager console provides the following ways to help you monitor operating system deployment objects.

Alerts for operating system deployments

You can configure an alert in the task sequence deployment settings to notify administrative users when compliance levels for the deployment is below the configured percentage.

After you configure the alert settings, if the specified conditions occur, Configuration Manager generates an alert. You can review task sequence deployment alerts at the following locations:

1. Review recent alerts in the **Operating Systems** node in the **Software Library** workspace.
2. Manage the configured alerts in the **Alerts** node in the **Monitoring** workspace.

Task sequence deployment status

After you deploy a task sequence, you can monitor the deployment status. Use the following procedure to monitor the deployment status for a task sequence.

To monitor deployment status

1. In the Configuration Manager console, click **Monitoring**.
2. In the Monitoring workspace, click **Deployments**.
3. Click the task sequence for which you want to monitor the deployment status.
4. On the **Home** tab, in the **Deployment** group, click **View Status**.

Operating system deployment reports

There are many predefined operating system deployment reports available. They are organized in several categories and can be used to report on specific information about state migration and task sequence deployments. In addition to using the preconfigured reports, you can also create custom software update reports according to the needs of your enterprise. For more information, see Operations and maintenance for reporting.

Monitor content

You can monitor content in the Configuration Manager console to review the status for all package types in relation to the associated distribution points. This can include the content validation status for the content in the package, the status of content assigned to a specific distribution point group, the state of content assigned to a distribution point, and the status of optional features for each distribution point (content validation, PXE, and multicast).

Content status monitoring

The **Content Status** node in the **Monitoring** workspace provides information about content packages. You can

review general information about the package, distribution status for the package, and detailed status information about the package. Use the following procedure to view content status.

To monitor content status

1. In the Configuration Manager console, click **Monitoring**.
2. In the Monitoring workspace, expand **Distribution Status**, and then click **Content Status**. The packages are displayed.
3. Select the package for which to view detailed status information.
4. On the **Home** tab, click **View Status**. Detailed status information for the package is displayed.

Distribution point group status

The **Distribution Point Group Status** node in the **Monitoring** workspace provides information about distribution point groups. You can review general information about the distribution point group, such as distribution point group status and compliance rate, as well as detailed status information for the distribution point group. Use the following procedure to view distribution point group status.

To monitor distribution point group status

1. In the Configuration Manager console, click **Monitoring**.
2. In the monitoring workspace, expand **Distribution Status**, and then click **Distribution Point Group Status**. The distribution point groups are displayed.
3. Select the distribution point group for which to view detailed status information.
4. On the **Home** tab, click **View Status**. Detailed status information for the distribution point group is displayed.

Distribution point configuration status

The **Distribution Point Configuration Status** node in the **Monitoring** workspace provides information about the distribution point. You can review which attributes are enabled for the distribution point, such as the PXE, Multicast, and content validation. You can also view detailed status information for the distribution point. Use the following procedure to view distribution point configuration status.

To monitor distribution point configuration status

1. In the Configuration Manager console, click **Monitoring**.
2. In the monitoring workspace, expand **Distribution Status**, and then click **Distribution Point Configuration Status**. The distribution points are displayed.
3. Select the distribution point for which to view distribution point status information.
4. In the results pane, click the **Details** tab. Status information for the distribution point is displayed.

Manage task sequences to automate tasks in System Center Configuration Manager

3/26/2017 • 26 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Use task sequences to automate steps in your System Center Configuration Manager environment. These steps can deploy an operating system image to a destination computer, build and capture an operating system image from a set of operating system installation files, and capture and restore user state information. Task sequences are located in the Configuration Manager console at **Software Library** > **Operating Systems** > **Task Sequence**. The **Task Sequence** node, including subfolders that you create, is replicated throughout the Configuration Manager hierarchy. For planning information, see Planning considerations for automating tasks.

Use the following sections to manage task sequences.

Create task sequences

Create task sequences by using the Create Task Sequence Wizard. This wizard can create the following types of task sequences:

TASK SEQUENCE TYPE	MORE INFORMATION
Task sequence to install an operating system	This task sequence type creates the steps to install an operating system, as well as the option to migrate user data, include software updates, and install applications.
Task sequence to upgrade an operating system	This task sequence type creates the steps to upgrade an operating system, as well as the option to include software updates and install applications.
Task sequence to capture an operating system	This task sequence type creates the steps to build and capture an operating system from a reference computer. You can include software updates and install applications on the reference computer before the image is captured.
Task sequence to capture and restore user state	This task sequence provides the steps to add to an existing task sequence to capture and restore user state data.
Task sequence to manage virtual hard disks	This task sequence type contains the steps to create a VHD, which includes to install an operating system and applications, that you can publish to System Center Virtual Machine Manager (VMM) from the Configuration Manager console.
Custom task sequence	This task sequence type does not add any steps to the task sequence. You must edit the task sequence and add steps to the task sequence after it is created.

Return to previous page when a task sequence fails

Beginning in Configuration Manager version 1702, you can return to a previous page when you run a task sequence and there is a failure. Prior to this release, you had to restart the task sequence when there was a failure.

For example, you can use the **Previous** button in the following scenarios:

- When a computer starts in Windows PE, the task sequence bootstrap dialog might display before the task sequence is available. When you click Next in this scenario, the final page of the task sequence displays with a message that there are no task sequences available. Now, you can click **Previous** to search again for available task sequences. You can repeat this process until the task sequence is available.
- When you run a task sequence, but dependent content packages are not yet available on distribution points, the task sequence fails. You can now distribute the missing content (if it wasn't distributed yet) or wait for the content to be available on distribution points, and then click **Previous** to have the task sequence search again for the content.

Edit a task sequence

You can modify a task sequence by adding or removing task sequence steps, adding or removing task sequence groups, or by changing the order of the steps. Use the following procedure to modify an existing task sequence.

> **IMPORTANT**
> When you edit a task sequence that was created by using the Create Task Sequence Wizard, the name of the step can be the action of the step or the type of the step. For example, you might see a step that has the name "Partition disk 0", which is the action for a step of type Format and Partition Disk. All task sequence steps are documented by their type, not necessarily by the name of the step that is displayed in the Editor.

To edit a task sequence

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. In the **Task Sequence** list, select the task sequence that you want to edit.
4. On the **Home** tab, in the **Task Sequence** group, click **Edit**, and then perform any of the following operations:
   - To add a task sequence step, click **Add**, select the type of the step, and then click the task sequence step that you want to add. For example, to add the Run Command Line step click **Add**, select **General**, and then click **Run Command Line**.

     For a list of all task sequence steps and their type, see the table that follows this procedure.
   - To add a group to the task sequence, click **Add**, and then click **New Group**. After you add a group you can then add steps to the group.
   - To change the order of the steps and groups in the task sequence, select the step or group that you want to re-order, and then use the **Move Item Up** or **Move Item Down** icons. You can move only one step or group at a time.
   - To remove a step or group, select the step or group and click **Remove**.
5. Click **OK** to save the changes.

For a list of the available task sequence steps, see Task sequence steps.

Configure high-impact task sequence settings

Beginning in Configuration Manager version 1702, you can set a task sequence as high-impact and customize the messages that users receive when they run the task sequence.

Set a task sequence as a high-impact task sequence

Use the following procedure to set a task sequence as high-impact.

> **NOTE**
> Any task sequence that meets certain conditions is automatically defined as high-impact. For details, see Manage high-risk deployments.

1. In the Configuration Manager console, go to **Software Library** > **Operating Systems** > **Task Sequences**.
2. Select the task sequence to edit, and click **Properties**.
3. On the **User Notification** tab, select **This is a high-impact task sequence**.

Create a custom notification for high-risk deployments

Use the following procedure to create a custom notification for high-impact deployments.

1. In the Configuration Manager console, go to **Software Library** > **Operating Systems** > **Task Sequences**.
2. Select the task sequence to edit, and click **Properties**.
3. On the **User Notification** tab, select **Use custom text**.

> **NOTE**
> You can only set user notification text when the **This is a high-impact task sequence** is selected.

4. Configure the following settings (max of 255 characters for each text box):

User notification headline text: Specifies the blue text that displays on the Software Center user notification. For example, in the default user notification, this section contains something like "Confirm you want to upgrade the operating system on this computer".

User notification message text: There are three text boxes that provide the body of the custom notification. All text boxes require that you add text.

- 1st text box: Specifies the main body of text, typically containing instructions for the user. For example, in the default user notification, this section contains something like "Upgrading the operating system will take time and your computer might restart several times."
- 2nd text box: Specifies the bold text under the main body of text. For example, in the default user notification, this section contains something like "This in-place upgrade installs the new operating system and automatically migrates your apps, data, and settings."
- 3rd text box: Specifies the last line of text under the bold text. For example, in the default user notification, this section contains something like "Click Install to begin. Otherwise, click Cancel."

Let's say you configure the following custom notification in properties.

```
![Custom notification for a task sequence](..\media\user-notification.png)

The following notification message will be displayed when the end-user opens the installation from Software
Center.

![Custom notification for a task sequence](..\media\user-notification-enduser.png)
```

Configure Software Center properties

Use the following procedure to configure the details for the task sequence displayed in Software Center. These details are for information only.

1. In the Configuration Manager console, go to **Software Library** > **Operating Systems** > **Task Sequences**.

2. Select the task sequence to edit, and click **Properties**.
3. On the **General** tab, the following settings for Software Center are available:

- **Restart required**: Lets the user know whether a restart is required during the installation.
- **Download size (MB)**: Specifies how many megabytes is displayed in Software Center for the task sequence.
- **Estimated run time (minutes)**: Specifies the estimated run time in minutes that's displayed in Software Center for the task sequence.

Distribute content referenced by a task sequence

Before clients run a task sequence that references content, you must distribute that content to distribution points. At any time, you can select the task sequence and distribute its content to build a new list of reference packages for distribution. If you make changes to the task sequence with updated content, you must redistribute the content before it is available to clients. Use the following procedure to distribute the content that is referenced by a task sequence.

To distribute referenced content to distribution points

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. In the **Task Sequence** list, select the task sequence that you want to distribute.
4. On the **Home** tab, in the **Deployment** group, click **Distribute Content** to start the Distribute Content Wizard.
5. On the **General** page, verify that the correct task sequence is selected for distribution, and then click **Next**.
6. On the **Content** page, verify the content to distribute, such as the boot image referenced by the task sequence, and then click **Next**.
7. On the **Content Destination** page, specify the collections, distribution point, or destination point group where you want to distribute the task sequence contents, and then click **Next**.

   > **IMPORTANT**
   > If the task sequence that you selected references content that is already distributed to a specific distribution point, that distribution point is not listed by the wizard.

8. Complete the wizard.

You can prestage the content referenced in the task sequence. Configuration Manager creates a compressed, prestaged content file that contains the files, associated dependencies, and associated metadata for the content that you select. Then, you can then manually import the content at a site server, secondary site, or distribution point. For more information about how to prestage content files, see Prestage content.

Deploy a task sequence

Use the following procedure to deploy a task sequence to the computers in a collection.

WARNING

You can manage the behavior for high-risk task sequence deployments. A high-risk deployment is a deployment that is automatically installed and has the potential to cause unwanted results. For example, a task sequence that has a purpose of **Required** that deploys an operating system is considered a high-risk deployment. For more information, see Settings to manage high-risk deployments.

NOTE

The status messages for the task sequence deployment are displayed in the Message window on a primary site, but they are not displayed on a central administration site.

To deploy a task sequence

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. In the **Task Sequence** list, select the task sequence that you want to deploy.
4. On the **Home** tab, in the **Deployment** group, click **Deploy**.

   NOTE

   If **Deploy** is not available, the task sequence has a reference that is not valid. Correct the reference and then try to deploy the task sequence again.

5. On the **General** page, specify the following information, and then click **Next**.

   - **Task sequence**: Specify the task sequence that you want to deploy. By default, this box displays the task sequence that you selected.
   - **Collection**: Specify the collection that contains the computers that will run the task sequence.

     Do not deploy task sequences that install operating systems to inappropriate collections, such as the **All Systems** collection. Be sure that the collection that you select contains only those computers that you want to run the task sequence.

NOTE

When you deploy a high-risk deployment, such as an operating system, the **Select Collection** window displays only the custom collections that meet the deployment verification settings that are configured in the site's properties. High-risk deployments are always limited to custom collections, collections that you create, and the built-in **Unknown Computers** collection. When you create a high-risk deployment, you cannot select a built-in collection such as **All Systems**. Uncheck **Hide collections with a member count greater than the site's minimum size configuration** to see all custom collections that contain fewer clients than the configured maximum size. For more information, see Settings to manage high-risk deployments.

The deployment verification settings are based on the current membership of the collection. After you deploy the task sequence, the collection membership is not reevaluated for the high-risk deployment settings.

For example, let's say you set **Default size** to 100 and the **Maximum size** to 1000. When you create a high risk deployment, the **Select Collection** window will only display collections that contain less than 100 clients. If you clear the **Hide collections with a member count greater than the site's minimum size configuration** setting, the window will display collections that contain less than 1000 clients.

When you select a collection that contains a site role, the following applies:

- If the collection contains a site system server and in the deployment verification settings you configure to block collections with site system servers, then an error occurs and you cannot continue.
  - If the collection contains a site system server and in the deployment verification settings you configure to warn you if collections that have site system servers, if the collection exceeds the default size value, or if the collection contains a server, then the Deploy Software Wizard will display a high risk warning. You must agree to create a high risk deployment and an audit status message is created.

- **Comments (optional)**: Specify additional information that describes this deployment of the task sequence.

6. On the **Deployment Settings** page, specify the following information, and then click **Next**.

   - **Purpose**: From the drop-down list, choose one of the following options:

     - **Available**: If the task sequence is deployed to a user, the user sees the published task sequence in the Application Catalog and can request it on demand. If the task sequence is deployed to a device, the user will see it in the Software Center and can install it on demand.

     - **Required**: The task sequence is deployed automatically, according to the configured schedule. However, a user can track the task sequence deployment status (if it is not hidden) and install the task sequence before the deadline by using the Software Center.

   - **Deploy automatically according to schedule whether or not a user is logged on**: This option is not available when you deploy a task sequence.

   - **Send wake-up packets**: If the deployment purpose is set to **Required** and this option is selected, a wake-up packet will be sent to computers before the deployment is installed to wake the computer from sleep at the installation deadline time. Before you can use this option, computers and networks must be configured for Wake On LAN.

   - **Allow clients on a metered Internet connection to download content after the installation deadline, which might incur additional costs**: When you have a task sequence that installs an application but does not deploy an operating system, you can specify whether to allow clients to download content after an installation deadline when they use metered Internet connections. Internet providers sometimes charge by the amount of data that you send and receive when you are on a metered Internet connection.

> **NOTE**
> While using a metered Internet connection might work for task sequences that do not deploy an operating system, it is not supported.

- **Require administrator approval if users request this application**: This option is not available when you deploy a task sequence.
- **Make available to the following**: Specify whether the task sequence is available to Configuration Manager clients, media, or PXE.

> **IMPORTANT**
> Use the **Only media and PXE (hidden)** setting for automated task sequence deployments. Select **Allow unattended operating system deployment** and set the SMSTSPreferredAdvertID variable as part of the media to have the computer automatically boot to the deployment with no user interaction. For more information about task sequence variables, see Task sequence built-in variables

7. On the **Scheduling** page, specify the following information, and then click **Next**.

> **IMPORTANT**
> When a Windows PE client starts from PXE or boot media, the client does not evaluate deployment schedules (start, expire, or deadline times). Only configure schedules in deployments to clients that start from the full Windows operating system. Consider using other methods, such as maintenance windows, to control active task sequences deployed to clients that start from Windows PE.

- **Schedule when this deployment will become available**: Specify the date and time when the task sequence is available to run on the destination computer. When you select the **UTC** check box, this setting ensures that the task sequence is available for multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

  If the start time is earlier than the required time, the client downloads the task sequence at the start time that you specify.

- **Schedule when this deployment will expire**: Specify the date and time when the task sequence expires on the destination computer. When you select the **UTC** check box, this setting ensures that the task sequence expires on multiple destination computers at the same time rather than at different times, according to the local time on the destination computers.

- **Assignment schedule**: Specify when the required task sequence is run on the destination computer. You can add multiple schedules.

  You can specify the date and time when the schedule starts, whether the task sequence runs weekly, monthly, or on a custom interval, and if the task sequence runs after an event such as logging on or logging off the computer.

> **NOTE**
> If you schedule a start time for a required task sequence that is earlier than the date and time when the task sequence is available, the Configuration Manager client downloads the task sequence at the scheduled start time, even though the task sequence is available at an earlier time.

- **Rerun behavior**: Specify when the task sequence is rerun. You can specify one of the following

options.

- **Never rerun deployed program**: The task sequence does not rerun on the client if the task sequence has been previously run on the client. The task sequence does not rerun even if it originally failed or if the task sequence files have been changed.
- **Always rerun program**: The task sequence is always rerun on the client when the deployment is scheduled, even if the task sequence has successfully run previously. This setting is particularly useful when you use recurring deployments in which the task sequence is routinely updated.

  > **IMPORTANT**
  > Although this option is set by default, it has no affect until you assign a required deployment. Available deployments can always be rerun by a user.

- **Rerun if failed previous attempt**: The task sequence is rerun when the deployment is scheduled only if the task sequence failed to run previously. This setting is particularly useful for required deployments so that they will automatically retry to run according to the assignment schedule if the last attempt to run was unsuccessful.
- Rerun if succeeded on previous attempt: The task sequence is rerun only if it has previously run successfully on the client. This setting is useful when you use recurring deployments in which the task sequence is routinely updated, and each update requires that the previous update is installed successfully.

> **NOTE**
> Because a user can rerun an available task sequence deployment, make sure that before you deploy an available task sequence in a product environment, you carefully evaluate and test what happens if a user reruns the task sequence multiple times.

8. On the **User Experience** page, specify the following information, and then click **Next**.

   - **Allow user to run the program independently of assignments**: Specify whether the user is allowed to run a required task sequence independently from the deployment assignments.
   - **Show Task Sequence progress**: Specify whether the Configuration Manager client displays the progress of the task sequence.
   - **Software installation**: Specify whether the user is allowed to install software outside a configured maintenance windows after the scheduled time.
   - **System restart (if required to complete the installation)**: Specify whether the user is allowed to restart the computer after a software installation outside a configured maintenance window after the assignment time.
   - **Allow task sequence to run for client on the Internet**: Specify whether the task sequence is allowed to run on an Internet-based client that Configuration Manager detects to be on the Internet. Operations that install software, such as an operating system, are not supported with this setting. Use this option only for generic script-based task sequences that perform operations in the standard operating system.

9. On the **Alerts** page, specify the alert settings that you want for this task sequence deployment, and then click **Next**.

10. On the **Distribution Points** page, specify the following information, and then click **Next**.

    - **Deployment options**: Specify one of the following options:

      > **NOTE**
      > When you use multicast to deploy an operating system the content must be downloaded to the destination computers either as it is needed or before the task sequence is run.

      - Specify that clients download content from the distribution point to the destination computer as it is needed by the task sequence.
      - Specify that clients download all the content from the distribution point to the destination computer before the task sequence is run. This option is not shown if you specified that the task sequence is available to PXE and boot media deployments (see the **Deployment Settings** page).
      - Specify that clients run the content from the distribution point. This option is available only when all packages associated with the task sequence is enabled to use a package share on the distribution point. To enable content to use a package share, see the **Data Access** tab in the **Properties** for each package.

    - **When no local distribution point is available, use a remote distribution point**: Specify whether clients can use distribution points that are on slow and unreliable networks to download the content that is required by the task sequence.

11. Complete the wizard.

Export and import task sequences

You can export and import task sequences with or without their related objects, such as such an operating system image, a boot image, a client agent package, a driver package, and applications that have dependencies.

Consider the following when you export and import task sequences.

- Passwords that are stored in the task sequence are not exported. If you export and import a task sequence that contains passwords, you must edit the imported task sequence and specify any passwords again. Ensure that you specify passwords for Join Domain or Workgroup, Connect To Network Folder, and Run Command Line actions.
- When you export a task sequence with the **Set Dynanmic Variables** step, no values are exported for variables that are configured with the **Secret value** setting. You must reenter the values for these variables after you import the task sequence.
- As a best practice, when you have multiple primary sites, import task sequences at the central administration site.

Use the following procedures to export and import a task sequence.

To export task sequences

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. In the **Task Sequence** list, select the task sequences that you want to export. If you select more than one task sequence, they are stored in one export file.
4. On the **Home** tab, in the **Task Sequence** group, click **Export** to start the Export Task Sequence Wizard.

5. On the **General** page, specify the following settings, and then click **Next**.

   - In the **File** box, specify the location and name of the export file. If you enter the file name directly, be sure to include the .zip extension to the file name. If you browse for the export file, the wizard automatically adds this file name extension.
   - Clear the **Export all task sequence dependencies** check box if you do not want to export task sequence dependencies. By default, the wizard scans for all the related objects and exports them with the task sequence. This includes any dependencies for applications.
   - Clear the **Export all content for the selected task sequences and dependencies** check box if you do not want to copy the content from the package source to the export location. If this check box is selected, the Import Task Sequence Wizard uses the import path as the new package source location.
   - In the **Administrator comments** box, add a description of the task sequences to export.

6. Complete the wizard.

The wizard creates the following output files:

- If you do not export content: a .zip file.
- If you export content: a .zip file and a folder named *export*_files, where *export* is the name of the .zip file that contains the exported content.

If you include content when you export a task sequence, make sure that you copy the .zip file and the *export*_files folder, or your import will fail.

To import task sequences

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Import Task Sequence** to start the Import Task Sequence Wizard.
4. On the **General** page, specify the exported .zip file, and then click **Next**.
5. On the **File Content** page, select the action that you require for each object that you import. This page shows all the objects that Configuration Manager will import.

   - If the object has never been imported, select **Create New**.
   - If the object has been previously imported, select one of the following actions:

     - **Ignore Duplicate** (default): This action does not import the object. Instead, the wizard links the existing object to the task sequence.
     - **Overwrite**: This action overwrites the existing object with the imported object. For applications, you can add a revision to update the existing application or create a new application.

6. Complete the wizard.

After you import the task sequence, edit the task sequence to specify any passwords that were in the original task sequence. For security reasons, passwords are not exported.

Create task sequence variables for computers and collections

You can define custom task sequence variables for computers and collections. Variables that are defined for a

computer are referred to as per-computer task sequence variables. Variables defined for a collection are referred to as per-collection task sequence variables. If there is a conflict, per-computer variables take precedence over per-collection variables. This means that task sequence variables that are assigned to a specific computer automatically have a higher priority than variables that are assigned to the collection that contains the computer.

For example, if collection ABC has a variable assigned to it and computer XYZ, which is a member of collection ABC, has a variable with the same name assigned to it, the variable that is assigned to computer XYZ has higher priority than that of the variable that is assigned to collection ABC.

You can hide per-computer and per-collection variables so that they are not visible in the Configuration Manager console. If you no longer want these variables to be hidden, you must delete them and redefine them without selecting the option to hide them. When you use the option **Do not display this value in the Configuration Manager console**, the value of the variable is not displayed, but can still be used by the task sequence when it runs.

You can manage per-computer variables at a primary site or at a central administration site. Configuration Manager does not support more than 1,000 assigned variables for a computer.

> **WARNING**
> When you use per-collection variables for task sequences, consider the following:
>
> - Because changes to collections are always replicated throughout the hierarchy, any changes that you make to collection variables will apply to not just members of the current site but to all members of the collection throughout the hierarchy.
>   - When you delete a collection, this action also deletes the task sequence variables that are configured for the collection.

Use the following procedures to create task sequence variables for a computer or collection.

To create task sequence variables for a computer

1. In the Configuration Manager console, click **Assets and Compliance**.
2. In the **Assets and Compliance** workspace, expand the collection that contains the computer that you want to add the variable to.
3. Select the computer and click **Properties**.
4. In the **Properties** dialog box, click the **Variables** tab.
5. For each variable that you want to create, click the **New** icon in the **Variable** dialog box and specify the name and the value of the task sequence variable. Clear the **Do not display this value in the Configuration Manager console** check box if you want to hide the variables so that they are not visible in the Configuration Manager console.
6. After you have added all the variables to the computer, click **OK**.

To create task sequence variables for a collection

1. In the Configuration Manager console, click **Assets and Compliance**.
2. In the **Assets and Compliance** workspace, select the collection that you want to add the variable to and click **Properties**.
3. In the **Properties** dialog box, click the **Collection Variables** tab.
4. For each variable that you want to create, click the **New** icon In the **Variable** dialog box and specify the name and the value of the task sequence variable. Clear the **Do not display this value in the Configuration Manager console** check box if you want to hide the variables so that they are not visible in the Configuration Manager console.

5. Optionally, specify the priority for Configuration Manager to use when the task sequence variables are evaluated.

6. After you have added all the variables to the collection, click **OK**.

Additional actions to manage task sequences

You can manage task sequences by using additional actions when you select the task sequence by using the following procedure.

To select a task sequence to manage

1. In the Configuration Manager console, click **Software Library**.

2. In the **Software Library** workspace, expand **Operating Systems,** and then click **Task Sequences**.

3. In the **Task Sequence** list, select the task sequence that you want to manage, and then select one of the available options.

Use the following table for more information about some of the additional actions to manage task sequences.

ACTION	DESCRIPTION
Copy	Makes a copy of the selected task sequence. You might find this action useful when you want to create a new task sequence that is based on an existing task sequence. When you make a copy of a task sequence in a folder, the copy is listed in that folder until you refresh the task sequence node. After the refresh, the copy appears in the root folder.
Disable	Disables the task sequence so that it cannot run on computers. Disabled task sequences can be deployed to computers, but computers do not run the task sequence until it is enabled.
Enable	Enables the task sequence so that it can be run. You do not need to redeploy a deployed task sequence after it is enabled.
Create Prestaged Content File	Starts the Create Prestaged Content File Wizard to prestage the task sequence content. For information about how to create a prestaged content file, see Prestage content.
Move	Moves the selected task sequence to another folder.
Properties	Opens the **Properties** dialog box for the selected task sequence. Use this dialog box to change the behavior of the task sequence object. However, you cannot change the steps of the task sequence by using this dialog box.

Next steps

Scenarios to deploy enterprise operating systems

Create a task sequence to install an operating system in System Center Configuration Manager

11/23/2016 • 10 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Use task sequences in System Center Configuration Manager to automatically install an operating system image on a destination computer. You create a task sequence that references a boot image used to start the destination computer, the operating system image that you want to install on the destination computer, and any other additional content, such as other applications or software updates, that you want to install. Then you deploy the task sequence to a collection that contains the destination computer.

Create a task sequence to install an operating system

There are a lot of scenarios to deploy an operating system to computers in your environment. In most cases, you will create a task sequence and select **Install an existing image package** in the Create Task Sequence Wizard to install the operating system, migrate user settings, apply software updates, and install applications. Before you create a task sequence to install an operating system, the following must be in place:

- **Required**
  - The boot image must be available in the Configuration Manager console.
  - An operating system image must be available in the Configuration Manager console.
- **Required (if used)**
  - Software updates must be synchronized in the Configuration Manager console.
  - Applications must be added to the Configuration Manager console.

To create a task sequence that installs an operating system

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, click **Install an existing Image package**, and then click **Next**.
5. On the **Task Sequence Information** page, specify the following settings, and then click **Next**.
   - **Task sequence name**: Specify a name that identifies the task sequence.
   - **Description**: Specify a description of the task that is performed by the task sequence.
   - **Boot image**: Specify the boot image that installs the operating system on the destination computer. The boot image contains a contain a version of Windows PE that is used to install the operating system, as well as any additional device drivers that are required. For information, see Manage boot images.

> **IMPORTANT**
> The architecture of the boot image must be compatible with the hardware architecture of the destination computer.

6. On the **Install Windows** page, specify the following settings, and then click **Next**.

   - **Image package**: Specify the package that contains the operating system image to install. For more information, see Manage operating system images.
   - **Image**: If the operating system image package has multiple images, specify the index of the operating system image to install.
   - **Partition and format the target computer installing the operating system**: Specify whether you want the task sequence to partition and format the destination computer before the operating system is installed.
   - **Product key**: Specify the product key for the Windows operating system to install. You can specify encoded volume license keys and standard product keys. If you use a non-encoded product key, each group of 5 characters must be separated by a dash (-). For example: *XXXXX-XXXXX-XXXXX-XXXXX-XXXXX*
   - **Server licensing mode**: Specify that the server license is **Per seat**, **Per server**, or that no license is specified. If the server license is **Per server**, also specify the maximum number of server connections.
   - Specify how to handle the administrator account that is used when the operating system image is deployed.
     - **Disable local administrator account**: Specify whether the local administrator account is disabled when the operating system image is deployed.
     - **Always use the same administrator password**: Specify whether the same password is used for the local administrator account on all computers where the operating system image is deployed.

7. On the **Configure Network** page, specify the following settings, and then click **Next**.

   - **Join a workgroup**: Specify whether to add the destination computer to a workgroup.
   - **Join a domain**: Specify whether to add the destination computer to a domain. In **Domain**, specify the name of the domain.

     > **IMPORTANT**
     > You can browse to locate domains in the local forest, but you must specify the domain name for a remote forest.

     You can also specify an organizational unit (OU). This is an optional setting that specifies the LDAP X.500-distinguished name of the OU in which to create the computer account if it does not already exist.
   - **Account**: Specify the user name and password for the account that has permissions to join the specified domain. For example: *domain\user* or *%variable%*.

> **IMPORTANT**
> You must enter the appropriate domain credentials if you plan to migrate either the domain settings or the workgroup settings.

8. On the **Install Configuration Manager** page, specify the Configuration Manager client package to install on the destination computer, and then click **Next**.
9. On the **State Migration** page, specify the following information, and then click **Next**.
   - **Capture user settings**: Specify whether the task sequence captures the user state. For more information about how to capture and restore the user state, see Manage user state.
   - **Capture network settings**: Specify whether the task sequence captures network settings from the destination computer. You can capture the membership of the domain or workgroup in addition to the network adapter settings.
   - **Capture Microsoft Windows settings**: Specify whether the task sequence captures Windows settings from the destination computer before the operating system image is installed. You can capture the computer name, registered user and organization name, and the time zone settings.
10. On the **Include Updates** page, specify whether to install required software updates, all software updates, or no software updates, and then click **Next**. If you specify to install software updates, Configuration Manager installs only those software updates that are targeted to the collections that the destination computer is a member of.
11. On the **Install Applications** page, specify the applications to install on the destination computer, and then click **Next**. If you specify multiple applications, you can also specify that the task sequence continues if the installation of a specific application fails.
12. Complete the wizard.

You can now deploy the task sequence to a collection of computers. For more information, see Deploy a task sequence.

Example task sequence to install an existing operating system image

Use the following table as a guide as you create a task sequence that deploys an operating system using an existing operating system image. The table will help you decide the general sequence for your task sequence steps and how to organize and structure those task sequence steps into logical groups. The task sequence that you create may vary from this sample and can contain more or less task sequence steps and groups.

> **IMPORTANT**
> You must always use the Create Task Sequence Wizard to create this task sequence.

When you use the Create Task Sequence Wizard to create this new task sequence some of the task sequence step names are different than what than what they would be if you manually added these task sequence steps to an existing task sequence. The following table displays the naming differences:

CREATE TASK SEQUENCE WIZARD TASK SEQUENCE STEP NAME	EQUIVALENT TASK SEQUENCE EDITOR STEP NAME
Request User State Storage	Request State Store
Capture User Files and Settings	Capture User State

CREATE TASK SEQUENCE WIZARD TASK SEQUENCE STEP NAME	EQUIVALENT TASK SEQUENCE EDITOR STEP NAME
Release User State Storage	Release State Store
Restart in Windows PE	Reboot to Windows PE or hard disk
Partition Disk 0	Format and Partition Disk
Restore User Files and Settings	Restore User State

TASK SEQUENCE GROUP OR STEP	DESCRIPTION
Capture File and Settings - **(New Task Sequence Group)**	Create a task sequence group. A task sequence group keeps similar task sequence steps together for better organization and error control. This group contains the steps needed to capture files and settings from the operating system of a reference computer.
Capture Windows Settings	Use this task sequence step to identify the Microsoft Windows settings to capture from the reference computer. You can capture the computer name, user and organizational information and the time zone settings.
Capture Network Settings	Use this task sequence step to capture network settings from the reference computer. You can capture the domain or workgroup membership of the reference computer and the network adapter setting information.
Capture User Files and Settings - **(New Task Sequence Sub-Group)**	Create a task sequence group within a task sequence group. This sub-group contains the steps needed to capture user state data. Similar to the initial group that you added, this sub-group keeps similar task sequence steps together for better organization and error control.
Request User State Storage	Use this task sequence step to request access to a state migration point where the user state data is stored. You can configure this task sequence step to capture or restore the user state information.
Capture User Files and Settings	Use this task sequence step to use the User State Migration Tool (USMT) to capture the user state and settings from the reference computer that will receive the task sequence associated with this task step. You can capture the standard options or configure whish options to capture.
Release User State Storage	Use this task sequence step to notify the state migration point that the capture or restore action is complete.
Install Operating System - **(New Task Sequence Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to install and configure the Windows PE environment.

TASK SEQUENCE GROUP OR STEP	DESCRIPTION
Restart in Windows PE	Use this task sequence step to specify the restart options for the destination computer that receives this task sequence. This step will display a message to the user indicating that the computer will be restarted so that the installation can continue. This step uses the read-only **_SMSTSInWinPE** task sequence variable. If the associated value equals **false** the task sequence step continues.
Partition Disk 0	This task sequence step specifies the actions necessary to format the hard drive on the destination computer. The default disk number is **0**. This step uses the read-only **_SMSTSClientCache** task sequence variable. This step will run if the Configuration Manager client cache does not exist.
Apply Operating System	Use this task sequence step to install the operating system image onto the destination computer. This step applies all volume images contained in the WIM file to the corresponding sequential disk volume on the target computer after first deleting all files on that volume (with the exception of Configuration Manager-specific control files). You can specify a **sysprep** answer file and also configure which disk partition is used for the installation.
Apply Windows Settings	Use this task sequence step to configure the Windows settings configuration information for the destination computer. The windows settings you can apply are user and organizational information, product or license key information, time zone, and the local administrator password.
Apply Network Settings	Use this task sequence step to specify the network or workgroup configuration information for the destination computer. You can also specify if the computer uses a DHCP server or you can statically assign the IP address information.
Apply Device Drivers	Use this task sequence step to install drivers as part of the operating system deployment. You can allow Windows Setup to search all existing driver categories by selecting **Consider drivers from all categories** or limit which driver categories Windows Setup searches by selecting **Limit driver matching to only consider drivers in selected categories**. This step uses the read-only **_SMSTSMediaType** task sequence variable. This task sequence step runs only if the value of the variable does not equal **FullMedia**.
Apply Driver Package	Use this task sequence step to make all device drivers in a driver package available for use by Windows setup.
Setup Operating System - **(New Task Sequence Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to set up the installed operating system.

TASK SEQUENCE GROUP OR STEP	DESCRIPTION
Setup Windows and ConfigMgr	Use this task sequence step to install the Configuration Manager client software. Configuration Manager installs and registers the Configuration Manager client GUID. You can assign the necessary installation parameters in the **Installation properties** window.
Install Updates	Use this task sequence step to specify how software updates are installed on the destination computer. The destination computer is not evaluated for applicable software updates until this task sequence step runs. At that point, the destination computer is evaluated for software updates similar to any other Configuration Manager-managed client. This step uses the read-only **_SMSTSMediaType** task sequence variable. This task sequence step runs only if the value of the variable does not equal **FullMedia**.
Restore User Files and Settings - **(New Task Sequence Sub-Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to restore the user files and settings.
Request User State Storage	Use this task sequence step to request access to a state migration point where the user state data is stored.
Restore User Files and Settings	Use this task sequence step to initiate the User State Migration Tool (USMT) to restore user state and settings to a destination computer.
Release User State Storage	Use this task sequence step to notify the state migration point that the user state data is no longer needed.

Create a task sequence to upgrade an operating system in System Center Configuration Manager

3/26/2017 • 8 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Use task sequences in System Center Configuration Manager to automatically upgrade an operating system from Windows 7 or later to Windows 10 on a destination computer. You create a task sequence that references the operating system image that you want to install on the destination computer and any other additional content, such as applications or software updates that you want to install. The task sequence to upgrade an operating system is part of the Upgrade Windows to the latest version scenario.

Create a task sequence to upgrade an operating system

To upgrade the operating system on computers to Windows 10, you can create a task sequence and select **Upgrade an operating system from upgrade package** in the Create Task Sequence Wizard. The wizard will add the steps to upgrade the operating system, apply software updates, and install applications. Before you create the task sequence, the following must be in place:

- **Required**
  - The Windows 10 operating system upgrade package must be available in the Configuration Manager console.
- **Required (if used)**
  - Software updates must be synchronized in the Configuration Manager console.
  - Applications must be added to the Configuration Manager console.

To create a task sequence that upgrades an operating system

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, click **Upgrade an operating system from upgrade package**, and then click **Next**.
5. On the **Task Sequence Information** page, specify the following settings, and then click **Next**.
   - **Task sequence name**: Specify a name that identifies the task sequence.
   - **Description**: Specify a description of the task that is performed by the task sequence.
6. On the **Upgrade the Windows Operating System** page, specify the following settings, and then click **Next**.
   - **Upgrade package**: Specify the upgrade package that contains the operating system upgrade source files. You can verify that you have selected the correct upgrade package by looking at the information in the **Properties** pane. For more information, see Manage operating system upgrade packages.

- **Edition index**: If there are multiple operating system edition indexes available in the package, select the desired edition index. By default, the first item is selected.

- **Product key**: Specify the product key for the Windows operating system to install. You can specify encoded volume license keys and standard product keys. If you use a non-encoded product key, each group of 5 characters must be separated by a dash (-). For example: *XXXXX-XXXXX-XXXXX-XXXXX-XXXXX*. When the upgrade is for a volume license edition, the product key is not required. You only need a product key when the upgrade is for a retail Windows edition.

7. On the **Include Updates** page, specify whether to install required software updates, all software updates, or no software updates, and then click **Next**. If you specify to install software updates, Configuration Manager installs only those software updates that are targeted to the collections that the destination computer is a member of.

8. On the **Install Applications** page, specify the applications to install on the destination computer, and then click **Next**. If you specify multiple applications, you can also specify that the task sequence continues if the installation of a specific application fails.

9. Complete the wizard.

Configure pre-cache content

Beginning in version 1702, for available deployments and task sequences, you can choose to use the pre-cache feature to have clients download only relevant content before a user installs the content.

> **TIP**
> Introduced with version 1702, the pre-cache is a pre-release feature. To enable it, see Use pre-release features from updates.

For example, let's say you want to deploy a Windows 10 in-place upgrade task sequence, only want a single task sequence for all users, and have multiple architectures and/or languages. Prior to version 1702, if you create an available deployment, and then the user clicks **Install** in Software Center, the content downloads at that time. This adds additional time before the installation is ready to start. Also, all content referenced in the task sequence is downloaded. This includes the operating system upgrade package for all languages and architectures. If each is roughly 3 GB in size, the download package can be quite large.

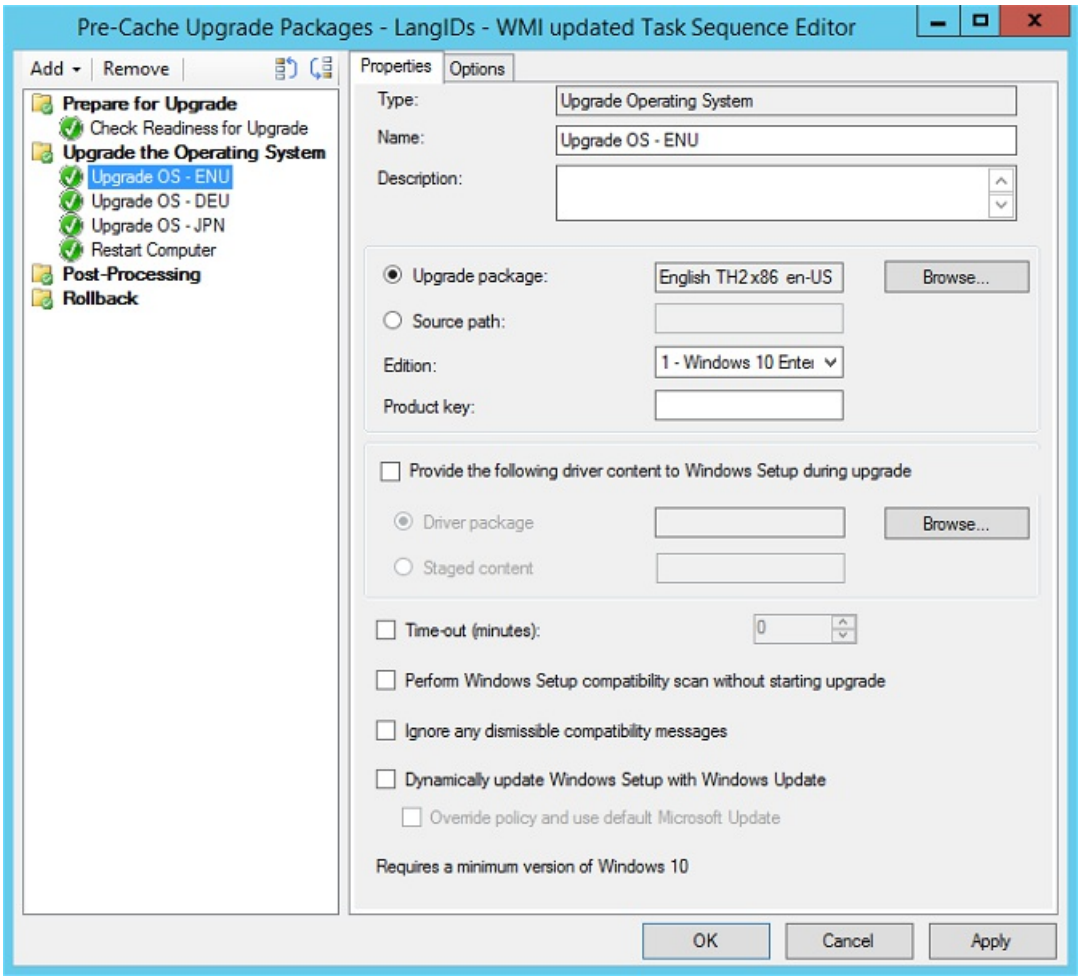
Pre-cache content gives you the option to allow the client to only download the applicable content as soon as it receives the deployment. Therefore, when the user clicks **Install** in Software Center, the content is ready and the installation starts quickly because the content is on the local hard drive.

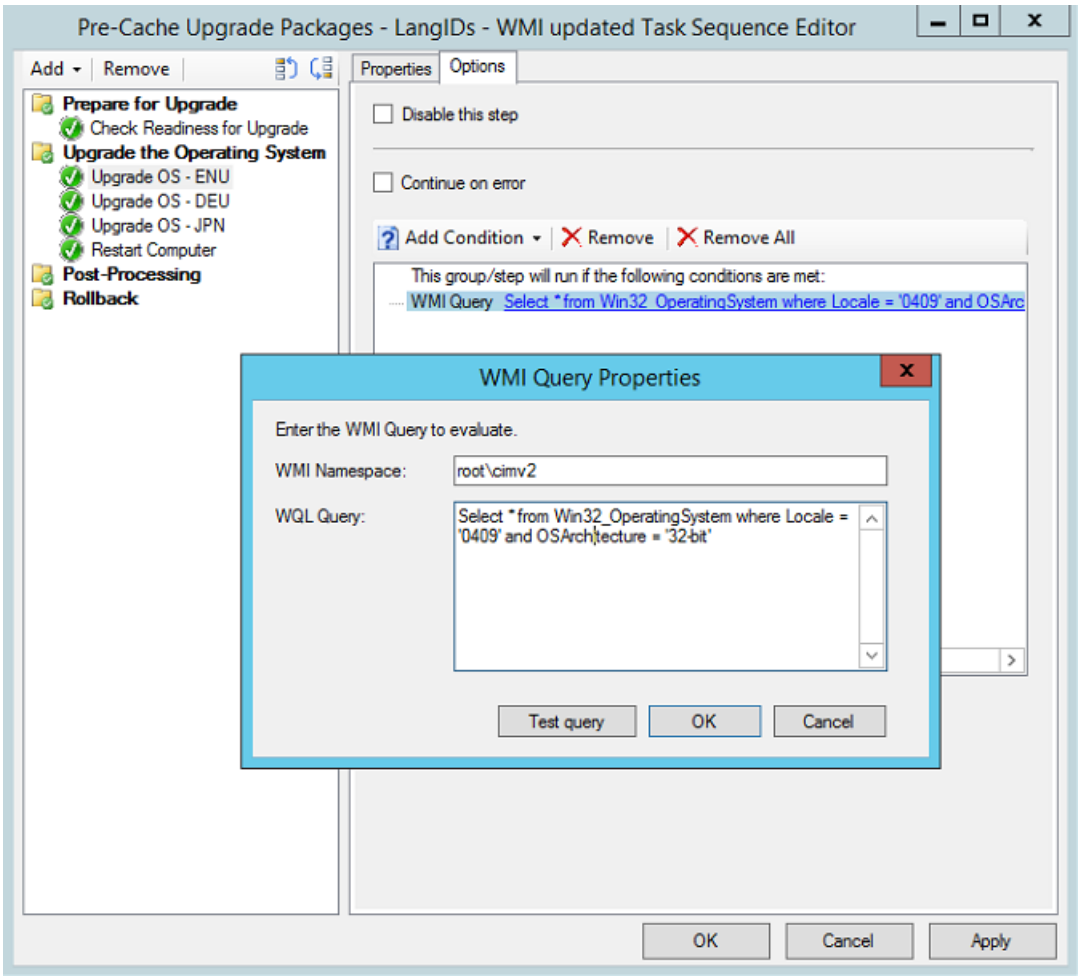
To configure the pre-cache feature

1. Create operating system upgrade packages for specific architectures and languages. Specify the architecture and language on the **Data Source** tab of the package. For the language, use the decimal conversion (for example, 1033 is the decimal for English and 0x0409 is the hexadecimal equivalent). For details, see Create a task sequence to upgrade an operating system.

   The architecture and language values are used to match task sequence step conditions that you will create in the next step to determine whether the operating system upgrade package should be pre-cached.

2. Create a task sequence with conditional steps for the different languages and architectures. For example, for the English version you could create a step like the following:





3. Deploy the task sequence. For the pre-cache feature, configure the following:

- On the **General** tab, select **Pre-download content for this task sequence**.
- On the **Deployment settings** tab, configure the task sequence with the **Available** for **Purpose**. If you create a **Required** deployment, the pre-cache functionality will not work.
- On the **Scheduling** tab, for the **Schedule when this deployment will be available** setting, choose a time in the future that gives clients enough time to pre-cache the content before the deployment is made available to users. For example, you can set the available time to be 3 hours in the future to allow enough time for the content to be pre-cached.
- On the **Distribution Points** tab, configure the **Deployment options** settings. If the content is not pre-cached on a client before a user starts the installation, these settings are used.

User experience

- When the client receives the deployment policy, it will start to pre-cache the content. This includes all referenced content (any other package types) and only the operating system upgrade package that matches the client based on the conditions that you set in the task sequence.
- When the deployment is made available to users (setting on the **Scheduling** tab of the deployment), a notification displays to inform users about the new deployment and the deployment becomes visible in Software Center. The user can go to Software Center and click **Install** to start the installation.
- If the content is not fully pre-cached, then it will use the settings specified on the **Deployment Option** tab of the deployment. We recommend that there is sufficient time between when the deployment is created and the time in which the deployment becomes available to users to allow clients enough time to pre-cache the content.

Download Package Content task sequence step

The Download Package Content step can be used before the **Upgrade Operating System** step in the following scenarios :

- You use a single upgrade task sequence that can work with both x86 and x64 platforms. To do this, include two **Download Package Content** steps in the **Prepare for Upgrade** group with conditions to detect the client architecture and download only the appropriate operating system upgrade package. Configure each **Download Package Content** step to use the same variable, and use the variable for the media path on the **Upgrade Operating System** step.

- To dynamically download an applicable driver package, use two **Download Package Content** steps with conditions to detect the appropriate hardware type for each driver package. Configure each **Download Package Content** step to use the same variable, and use the variable for the **Staged content** value in drivers section on the **Upgrade Operating System** step.

> **NOTE**
> When there is more than one package, Configuration Manager adds a numerical suffix to the variable name. For example, if you specify a variable of %mycontent% as a custom variable, this is the root for where all the referenced content is stored (which can be multiple packages). When you refer to the variable in a subsequence step, such as Upgrade Operating System, it is used with a numerical suffix. In this example, %mycontent01% or %mycontent02% where the number corresponds to the order in which the package is listed in this step.

Optional post-processing task sequence steps

After you create the task sequence, you can add additional steps to uninstall applications with known compatibility issues, or add post-processing actions to run after the computer is restarted and the upgrade to Windows 10 is successful. Add these additional steps in the Post-Processing group of the task sequence.

NOTE

Because this task sequence is not linear, there are conditions on steps that can affect the results of the task sequence, depending on whether it successfully upgrades the client computer or if it has to roll back the client computer to the operating system version it started with.

Optional rollback task sequence steps

When something goes wrong with the upgrade process after the computer is restarted, Setup will roll back the upgrade to the previous operating system and the task sequence will continue with any steps in the Rollback group. After you create the task sequence, you can add optional steps to the Rollback group.

Folder and files removed after computer restart

When the task sequence to upgrade an operating system to Windows 10 and all other steps in the task sequence are complete, the post-processing and rollback scripts are not removed until the computer is restarted. These script files do not contain sensitive information.

Create a task sequence to capture an operating system in System Center Configuration Manager

11/23/2016 • 14 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

When you use a task sequence to deploy an operating system to a computer in System Center Configuration Manager, the computer installs the operating system image that you specify in the task sequence. To customize the operating system image so it includes specific drivers, applications, software updates, etc., you use a build and capture task sequence to build a reference computer and then capture the operating system image from that reference computer. If you already have a reference computer available to capture, you can create a custom task sequence to capture the operating system. Use the following sections to capture a custom operating system.

Use a task sequence to build and capture a reference computer

The build and capture task sequence partitions and formats the reference computer, installs the operating system, as well as the Configuration Manager client, applications, and software updates, and then captures the operating system from the reference computer. The packages associated with the task sequence, such as applications, must be available on distribution points before you create the build and capture task sequence.

Prepare for operating system deployments

There are a lot of scenarios to deploy an operating system to computers in your environment. In most cases, you will create a task sequence and select **Install an existing image package** in the Create Task Sequence Wizard to install the operating system, migrate user settings, apply software updates, and install applications. Before you create a task sequence to install an operating system, the following must be in place:

- **Required**
  - The boot image must be available in the Configuration Manager console.
  - An operating system image must be available in the Configuration Manager console.
- **Required (if used)**
  - Driver packages that contain the necessary Windows drivers to support hardware on the reference computer must be available in the Configuration Manager console. For more information about the task sequence steps to manage drivers, see Use task sequences to install device drivers.
  - Software updates must be synchronized in the Configuration Manager console.
  - Applications must be added to the Configuration Manager console.

Create a build and capture task sequence

Use the following procedure to use a task sequence to build a reference computer and capture the operating system.

To create a task sequence that builds and captures an operating system image

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.

4. On the **Create a New Task Sequence** page, select **Build and capture a reference operating system image**.

5. On the **Task Sequence Information** page, specify the following settings, and then click **Next**.

   - **Task sequence name**: Specify a name that identifies the task sequence.
   - **Description**: Specify a description of the task that is performed by the task sequence, such as a description of the operating system that is created by the task sequence.
   - **Boot image**: Specify the boot image that installs the operating system image.

     > **IMPORTANT**
     > The architecture of the boot image must be compatible with the hardware architecture of the destination computer.

6. On the **Install Windows** page, specify the following settings, and then click **Next**.

   - **Image package**: Specify the operating system image package, which contains the files that are required to install the operating system.
   - **Image index**: Specify the operating system to install. If the operating system image contains multiple versions, select the version that you want to install.
   - **Product key**: Specify the product key for the Windows operating system to install. You can specify encoded volume license keys and standard product keys. If you use a non-encoded product key, each group of 5 characters must be separated by a dash (-). For example: *XXXXX-XXXXX-XXXXX-XXXXX-XXXXX*
   - **Server licensing mode**: Specify that the server license is **Per seat**, **Per server**, or that no license is specified. If the server license is **Per server**, also specify the maximum number of server connections.
   - Specify how to handle the administrator account that is used when the operating system is deployed.
     - **Randomly generate the local administrator password and disable the account on all supported platforms**: Specify whether to have Configuration Manager create a random password for the local administrator account and disable the account when the operating system is deployed.
     - **Enable the account and specify the local administrator password**: Specify whether the same password is used for the local administrator account on all computers where the operating system is deployed.

7. On the **Configure Network** page, specify the following settings, and then click **Next**.

   - **Join a workgroup**: Specify whether to add the destination computer to a workgroup when the operating system is deployed.
   - **Join a domain**: Specify whether to add the destination computer to a domain when the operating system is deployed. In **Domain**, specify the name of the domain.

     > **IMPORTANT**
     > You can browse to locate domains in the local forest, but you must specify the domain name for a remote forest.

     You can also specify an organizational unit (OU). This is an optional setting that specifies the LDAP

X.500-distinguished name of the OU in which to create the computer account if it does not already exist.

- **Account**: Specify the user name and password for the account that has permissions to join the specified domain. For example: *domain\user* or *%variable%*.

  > **IMPORTANT**
  > You must enter the appropriate domain credentials if you plan to migrate either the domain settings or the workgroup settings.

8. On the **Install Configuration Manager** page, specify the Configuration Manager client package that contains the source files to install the Configuration Manager client, add any additional properties needed to install the client, and then click **Next**.

   For more information about properties that can be used to install a client, see About client installation properties.

9. On the **Include Updates** page, specify whether to install required software updates, all software updates, or no software updates, and then click **Next**. If you specify to install software updates, Configuration Manager installs only those software updates that are targeted to the collections that the destination computer is a member of.

10. On the **Install Applications** page, specify the applications to install on the destination computer, and then click **Next**. If you specify multiple applications, you can also specify that the task sequence continues if the installation of a specific application fails.

11. On the **System Preparation** page, specify the following settings, and then click **Next**.

    - **Package**: Specify the Configuration Manager package that contains the appropriate version of Sysprep to use to capture the reference computer settings.

      If the operating system version that you are running is Windows Vista or later, Sysprep is automatically installed on the computer and you do not have to specify a package.

12. On the **Images Properties** page, specify the following settings for the operating system image, and then click **Next**.

    - **Created by**: Specify the name of the user who created the operating system image.
    - **Version**: Specify a user-defined version number that is associated with the operating system image.
    - **Description**: Specify a user-defined description of the operating system computer image.

13. On the **Capture Image** page, specify the following settings, and then click **Next**.

    - **Path**: Specify a shared network folder where the output .WIM file is stored. This file contains the operating system image that is based on the settings that you specify by using this wizard. If you specify a folder that contains an existing .WIM file, the existing file is overwritten.
    - **Account**: Specify the Windows account that has permissions to the network share where the image is stored.

14. Complete the wizard.

15. To add additional steps to the task sequence, select the task sequence that you created and click **Edit**. For information about how to edit a task sequence, see Edit a task sequence.

Deploy the task sequence to a reference computer in one of the following ways:

- If the reference computer is a Configuration Manager client, you can deploy the build and capture task sequence to the collection that contains the reference computer. For information about how to deploy the operating system image, see Create a task sequence to install an operating system.

  > **NOTE**
  > If the task sequence has a disk partitioning task sequence step, do not select the **Download Program** option when you deploy the task sequence.

- If the reference computer is not a Configuration Manager client or if you want to manually run the task sequence on the reference computer, run the **Create Task Sequence Media Wizard** to create bootable media. For information about how to create bootable media, see Create bootable media.

Capture an operating system image from an existing reference computer

When you already have a reference computer ready to capture, you can create a task sequence that captures the operating system from the reference computer. You will use the **Capture Operating System Image** task sequence step to capture one or more images from a reference computer and store them in a image file (.wim) on the specified network share. The reference computer is started in Windows PE by using a boot image, each hard drive on the reference computer is captured as a separate image within the .wim file. If the referenced computer has multiple drives, the resulting .wim file will contain a separate image for each volume. Only volumes that are formatted as NTFS or FAT32 are captured. Volumes with other formats and USB volumes are skipped.

Use the following procedure to capture an operating system image from an existing reference computer.

To capture an operating system from an existing reference computer

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, select **Create a new custom task sequence**.
5. On the **Task Sequence Information** page, specify a name for the task sequence and a description of the task sequence.
6. Specify a boot image for the task sequence. This boot image is used to start the reference computer with Windows PE. For more information, see Manage boot images.
7. Complete the wizard.
8. In **Task Sequences**, select the custom task sequence, and then on the **Home** tab, in the **Task Sequence** group, click **Edit** to open the task sequence editor.
9. Use this step only if the Configuration Manager client is installed on the reference computer.

   Click **Add**, click **Images**, and then click Prepare ConfigMgr Client for Capture. This task sequence step takes the Configuration Manager client on the reference computer and prepares it for capture as part of the imaging process.

10. Click **Add**, click **Images**, and then click Prepare Windows for Capture. This task sequence action runs Sysprep and then reboots the computer into Windows PE boot image specified for the task sequence. The reference computer must not be joined to a domain for this action to be completed successfully.
11. Click **Add**, click **Images**, and then click Capture Operating System Image. This task sequence step will only

run from Windows PE to capture the hard drives on the reference computer. Configure the following settings for the task sequence step.

- **Name** and **Description**: Optionally, you can change the name of the task sequence step and provide a description.
- **Destination**: Specify a shared network folder where the output .WIM file is stored. This file contains the operating system image that is based on the settings that you specify by using this wizard. If you specify a folder that contains an existing .WIM file, the existing file is overwritten.
- **Description**, **Version**, and **Created by**: Optionally, provide details about the image that you will capture.
- **Capture operating system image account**: Specify the Windows account that has permissions to the network share you specified. Click **Set** to specify the name of that Windows account.

Click **OK** to close the task sequence editor.

Deploy the task sequence to a reference computer in one of the following ways:

- If the reference computer is a Configuration Manager client, you can deploy the task sequence to the collection that contains the reference computer. For information about how to deploy the operating system image, see Create a task sequence to install an operating system.
- If the reference computer is not a Configuration Manager client or if you want to manually run the task sequence on the reference computer, run the **Create Task Sequence Media Wizard** to create bootable media. For information about how to create bootable media, see Create bootable media.

Task sequence example to build and capture an operating system image

Use the following table as a guide as you create a task sequence that builds and captures an operating system image. The table will help you decide the general sequence for your task sequence steps and how to organize and structure those task sequence steps into logical groups. The task sequence that you create may vary from this sample and can contain more or less task sequence steps and groups.

> **IMPORTANT**
> Always use the Create Task Sequence Wizard to create this type of task sequence.

When you use the **New Task Sequence** to create this new task sequence some of the task sequence step names are different than what they would be if you manually added these task sequence steps to an existing task sequence. The following table displays the naming differences:

NEW TASK SEQUENCE WIZARD TASK SEQUENCE STEP NAME	EQUIVALENT TASK SEQUENCE EDITOR STEP NAME
Restart in Windows PE	Reboot to Windows PE or hard disk
Partition Disk 0	Format and Partition Disk
Apply Device Drivers	Auto Apply Drivers
Install Updates	Install Software Updates
Join Workgroup	Join Domain or Workgroup

NEW TASK SEQUENCE WIZARD TASK SEQUENCE STEP NAME	EQUIVALENT TASK SEQUENCE EDITOR STEP NAME
Prepare ConfigMgr Client	Prepare ConfigMgr Client for Capture
Prepare Operating System	Prepare Windows for Capture
Capture the Reference Machine	Capture Operating System Image

TASK SEQUENCE GROUP/STEP	REFERENCE
Build the Reference Computer - **(New Task Sequence Group)**	Create a task sequence group. A task sequence group keeps similar task sequence steps together for better organization and error control. This group contains the actions necessary to build a reference computer.
Restart in Windows PE	Use this task sequence step to specify the restart options for the destination computer. This step will display a message to the user that the computer will be restarted so that the installation can continue. This step uses the read-only **_SMSTSInWinPE** task sequence variable. If the associated value equals **false** the task sequence step will continue.
Partition Disk 0	Use this task sequence step to specify the actions necessary to format the hard drive on the destination computer. The default disk number is **0**. This step uses the read-only **_SMSTSClientCache** task sequence variable. This step will run if the Configuration Manager client cache does not exist.
Apply Operating System	Use this task sequence step to install a specified operating system image on the destination computer. This step applies all volume images contained in the WIM file to the corresponding sequential disk volume on the target computer after first deleting all files on that volume (with the exception of Configuration Manager-specific control files).
Apply Windows Settings	Use this task sequence step to configure the Windows settings configuration information for the destination computer.
Apply Network Settings	Use this task sequence step to specify the network or workgroup configuration information for the destination computer.

TASK SEQUENCE GROUP/STEP	REFERENCE
Apply Device Drivers	You his task sequence step to match and install drivers as part of an operating system deployment. You can allow Windows Setup to search all existing driver categories by selecting **Consider drivers from all categories** or limit which driver categories Windows Setup searches by selecting **Limit driver matching to only consider drivers in selected categories**. This step uses the read-only **_SMSTSMediaType** task sequence variable. If the associated value does not equal **FullMedia** this task sequence step will run.
Setup Windows and ConfigMgr	Use this task sequence step to install the Configuration Manager client software. Configuration Manager installs and registers the Configuration Manager client GUID. You can assign the necessary installation parameters in the **Installation properties** window.
Install Updates	Use this task sequence step to specify how software updates are installed on the destination computer. The destination computer is not evaluated for applicable software updates until this task sequence step runs. At that point, the destination computer is evaluated for software updates similar to any other Configuration Manager-managed client. This step uses the read-only **_SMSTSMediaType** task sequence variable. If the associated value does not equal **FullMedia** this task sequence step will run.
Capture the Reference Computer - **(New Task Sequence Group)**	Create another a task sequence group. This group contains the necessary steps to prepare and capture a reference computer.
Join Workgroup	Use this task sequence step to specify information needed to have the destination computer join a workgroup.
Prepare ConfigMgr Client for Capture	Use this step to take the Configuration Manager client on the reference computer and prepares it for capture as part of the imaging process
Prepare Operating System	Use this task sequence step to specify the Sysprep options to use when capturing Windows settings from the reference computer. This task sequence step runs Sysprep and then reboots the computer into the Windows PE boot image specified for the task sequence.
Capture Operating System Image	Use this task sequence step to enter a specific existing network share and .WIM file to use when saving the image. This location is used as the package source location when adding an operating system image package using the **Add Operating System Image Package Wizard**.

After you have captured an image from a reference computer, do not capture another operating system image from the reference computer because registry entries are created during the initial configuration. Create a new reference computer each time that you capture the operating system image. If you plan to use the same reference computer to create future operating system images, first uninstall the Configuration Manager client, and then reinstall the Configuration Manager client.

Next steps

Methods to deploy enterprise operating systems

Create a task sequence to capture and restore user state in System Center Configuration Manager

11/23/2016 • 8 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can use System Center Configuration Manager task sequences to capture and restore the user state data in operating system deployment scenarios where you want to retain the user state of the current operating system. Depending on the type of task sequence you create, the capture and restore steps might be automatically added as part of the task sequence. In other scenarios, you might need to manually add the capture and restore steps to the task sequence. This topic provides the steps that you must add to an existing task sequence to capture and restore user state data.

How to capture and restore user state data

To capture and restore the user state, you must add the following steps to the task sequence:

- **Request State Store**: This step is needed only if you store the user state on the state migration point.
- **Capture User State**: This step captures the user state data and stores it on the state migration point or locally using links.
- **Restore User State**: This step restores the user state data on the destination computer. It can retrieve the data from a user state migration point or from the destination computer.
- **Release State Store**: This step is needed only if you store the user state on the state migration point. This step removes this data from the state migration point.

Use the following procedures to add the task sequence steps needed to capture the user state and restore the user state. For more information about creating a task sequences, see Manage task sequences to automate tasks.

To add task sequence steps to capture the user state

1. In the **Task Sequence** list, select a task sequence, and then click **Edit**.
2. If you are using a state migration point to store the user state, add the **Request State Store** step to the task sequence. In the **Task Sequence Editor** dialog box, click **Add**, point to **User State**, and then click **Request State Store**. Specify the following properties and options for the **Request State Store** step, and then click **Apply**.

   On the **Properties** tab, specify the following options:

   - Enter a name and description for the step.
   - Click **Capture state from the computer**.
   - In the **Number of retries** box, specify the number of times the task sequence attempts to capture the user state data if an error occurs.
   - In the **Retry delay (in seconds)** box, specify how many seconds that the task sequence waits before it retries to capture the data.
   - Select the **If computer account fails to connect to state store, use the Network Access account** check box to specify whether to use the Configuration Manager Network Access Account to connect to the state store.

On the **Options** tab, specify the following options:

- Select the **Continue on error** check box if you want the task sequence to continue to the next step if this step fails.
- Specify any conditions that must be met before the task sequence can continue if an error occurs.

3. Add the **Capture User State** step to the task sequence. In the **Task Sequence Editor** dialog box, click **Add**, point to **User State**, and then click **Capture User State**. Specify the following properties and options for the **Capture User State** step, and then click **OK**.

> **IMPORTANT**
> When you add this step to your task sequence, also set the **OSDStateStorePath** task sequence variable to specify where the user state data is stored. If you store the user state locally, do not specify a root folder as that can cause the task sequence to fail. When you store the user data locally always use a folder or subfolder. For information about this variable, see Capture User State Task Sequence Action Variables.

On the **Properties** tab, specify the following options:

- Enter a name and description for the step.
- Specify the package that contains the USMT source file used to capture the user state data.
- Specify the user profiles to capture:
  - Click **Capture all user profiles with standard options** to capture all user profiles.
  - Click **Customize user profile capture** to specify individual user profiles to capture.
- Select **Enable verbose logging** to specify how much information to write to log files if an error occurs.
- Select **Skip files that use the Encrypting File System (EFS)**.
- Select **Copy by using file system access** to specify the following settings:
  - **Continue if some files cannot be captured**: This setting allows the task sequence step to continue the migration process even if some files cannot be captured. If you disable this option and a file cannot be captured, the task sequence step fails. This option is enabled by default.
  - **Capture locally by using links instead of by copying files**: This setting allows you to use the hard link migration feature that is available in USMT 4.0. This setting is ignored if you use versions of USMT that are earlier than USMT 4.0.
  - **Capture in off-line mode (Windows PE only)**: This setting allows you to capture use state from Windows PE without booting to the existing operating system. This setting is ignored if you use versions of USMT that are earlier than USMT 4.0.
- Select **Capture by using Volume Copy Shadow Services (VSS)**. This setting is ignored if you use versions of USMT that are earlier than USMT 4.0.

On the **Options** tab, specify the following options:

- Select the **Continue on error** check box if you want the task sequence to continue to the next step if this step fails.
- Specify any conditions that must be met before the task sequence can continue if an error occurs.

4. If you are using a state migration point to store the user state, add the Release State Store step to the task sequence. In the **Task Sequence Editor** dialog box, click **Add**, point to **User State**, and then click **Release State Store**. Specify the following properties and options for the **Release State Store** step, and then click **OK**.

   > **IMPORTANT**
   > The task sequence action that runs before the **Release State Store** step must be successful before the **Release State Store** step is started.

   On the **Properties** tab, enter a name and description for the step.

   On the **Options** tab, specify the following options.

   - Select the **Continue on error** check box if you want the task sequence to continue to the next step if this step fails.
   - Specify any conditions that must be met before the task sequence can continue when an error occurs.

Deploy this task sequence to capture the user state on a destination computer. For information about how to deploy task sequences, see Deploy a task sequence.

To add task sequence steps to restore the user state

1. In the **Task Sequence** list, select a task sequence, and then click **Edit**.
2. Add the Restore User State step to the task sequence. In the **Task Sequence Editor** dialog box, click **Add**, point to **User State**, and then click **Restore User State**. This step establishes a connection to the state migration point. Specify the following properties and options for the **Restore User State** step, and then click **OK**.

   On the **Properties** tab, specify the following properties:

   - Enter a name and description for the step.
   - Specify the package that contains the USMT to restore the user state data.
   - Specify the user profiles to restore:
     - Click **Restore all captured user profiles with standard options** to restore all user profiles.
     - Click **Customize user profile capture** to restore individual user profiles.
   - Select **Restore local computer user profiles** to provide a new password for the restored profiles. You cannot migrate passwords for local profiles.

     > **NOTE**
     > When you have local user accounts, and you use the Capture User State step and select **Capture all user profiles with standard options**, you must select the **Restore local computer user profiles** setting in the Restore User State step or the task sequence will fail.

   - Select **Continue if some files cannot be restored** if you want the **Restore User State** step to continue if a file cannot be restored.

     If you store the user state by using local links and the restore is not successful, the administrative user can manually delete the hard-links that were created to store the data or the task sequence can run the USMTUtils tool. If you use USMTUtils to delete the hard-link, add a Restart Computer step

after you run USMTUtils.

- Select **Enable verbose logging** to specify how much information to write to log files if an error occurs.

On the **Options** tab, specify the following options:

- Select the **Continue on error** check box if you want the task sequence to continue to the next step if this step fails.
- Specify any conditions that must be met before the task sequence can continue if an error occurs.

3. If you are using a state migration point to store the user state, add the Release State Store step to the task sequence. In the **Task Sequence Editor** dialog box, click **Add**, point to **User State**, and then click **Release State Store**. Specify the following properties and options for the **Release State Store** step, and then click **OK**.

   > **IMPORTANT**
   > The task sequence action that runs before the **Release State Store** step must be successful before the **Release State Store** step is started.

   On the **Properties** tab, enter a name and description for the step.

   On the **Options** tab, specify the following options.

   - Select the **Continue on error** check box if you want the task sequence to continue to the next step if this step fails.
   - Specify any conditions that must be met before the task sequence can continue when an error occurs.

Deploy this task sequence to restore the user state on a destination computer. For information about deploying task sequences, see Deploy a task sequence.

Next steps

Monitor the task sequence deployment

Use a task sequence to manage virtual hard disks in System Center Configuration Manager

11/23/2016 • 16 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

In System Center Configuration Manager, you can manage virtual hard disks (VHDs) and integrate the VHDs that you create into your datacenter from the Configuration Manager console. Specifically, you can create and modify a VHD, add applications and software updates to the VHD, and publish the VHD to System Center Virtual Machine Manager (VMM) from the Configuration Manager console.

Use the following sections to manage VHDs in Configuration Manager.

Prerequisites

Verify the following prerequisites before you begin:

- The computer from which you manage VHDs must run one of the following operating systems:
  - Windows 8.1 x64
  - Windows 8 x64
  - Windows Server 2008 R2
  - Windows Server 2012
  - Windows Server 2012 R2
- Virtualization must be enabled in the BIOS and Hyper-V must be installed on the computer from which you run the Configuration Manager console to manage VHDs. Also as a best practice, install the Hyper-V management tools to help you test and troubleshoot your virtual hard disks. For example, to monitor the smsts.log file to track the progress of the task sequence in Hyper-V you must have the Hyper-V management tools installed. For more information about Hyper-V requirements, see Hyper-V Installation Prerequisites.

  > **IMPORTANT**
  > The process to create a VHD consumes processor time and memory. Therefore, it is recommended that you manage VHDs from a Configuration Manager console that is not installed on the site server.

- The site server must have **Write** access permission to the folder that will contain the VHD file when you manage VHDs from a computer that is remote from the site server.
- Verify that you have enough free disk space on the computer from which you manage the VHDs. The hard disk space requirements of the VHD will vary depending on the operating system and applications that you install.
- Verify that you have enough memory on the computer from which you manage the VHDs. During the process to create the VHD, the virtual machine is configured to consume 2 GB of memory.
- Install the System Center Virtual Machine Manager (VMM) console on the computer from which you upload the VHD to VMM. You can install the VMM console on a separate computer from which you manage your

VHDs, which means you do not need to have Hyper-V installed to import the VHD to VMM.

> **NOTE**
> If you install the VMM console while the Configuration Manager console is open, you must restart the Configuration Manager console after the VMM console installation completes. Otherwise, Configuration Manager will not successfully connect to the VMM management server to upload a VHD.

Steps to Create a VHD

To create a VHD, you must create a task sequence that contains the steps to create the VHD, and then use the task sequence in the Create Virtual Hard Drive Wizard to create the VHD. The following sections provide the steps to create the VHD.

Create a Task Sequence for the VHD

You must create a task sequence that will contain the steps to create the VHD. In the Create Task Sequence Wizard, you have the **Install an existing image package to a virtual hard disk** option that creates the steps to use to create the VHD. For example, the wizard adds the following required steps: Restart in Windows PE, Format and Partition Disk, Apply Operating System, and Shutdown Computer. You cannot create the VHD while in the full operating system. Also, Configuration Manager must wait until the virtual machine is shut down before it can complete the package. By default, the wizard waits for 5 minutes before it shuts down the virtual machine. After you create the task sequence you can add additional steps if necessary.

> **IMPORTANT**
> The following procedure creates the task sequence by using the **Install an existing image package to a virtual hard disk** option, which automatically includes the required steps to successfully create the VHD. If you choose to use an existing task sequence or manually create a task sequence, be sure that you add the Shutdown Computer step at the end of the task sequence. Without this step, the temporary virtual machine is not deleted and process to create the VHD does not complete. However, the wizard completes and reports success.

Use the following procedure to create the task sequence to create the VHD:

To create the task sequence to create the VHD

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, click **Install an existing image package to a virtual hard disk**, and then click **Next**.
5. On the **Task Sequence Information** page, specify the following settings, and then click **Next**.
   - **Task sequence name**: Specify a name that identifies the task sequence.
   - **Description**: Specify a description of the task sequence.
   - **Boot image**: Specify the boot image that installs the operating system on the destination computer. For more information, see Manage boot images.
6. On the **Install Windows** page, specify the following settings, and then click **Next**.
   - **Image package**: Specify the package that contains the operating system image to install.

- **Image**: If the operating system image package has multiple images, specify the index of the operating system image to install.
- **Product key**: Specify the product key for the Windows operating system to install. You can specify encoded volume license keys and standard product keys. If you use a non-encoded product key, each group of 5 characters must be separated by a dash (-). For example: *XXXXX-XXXXX-XXXXX-XXXXX-XXXXX*
- **Server licensing mode**: Specify that the server license is **Per seat**, **Per server**, or that no license is specified. If the server license is **Per server**, also specify the maximum number of server connections.
- Specify how to handle the administrator account that is used when the operating system image is deployed.
  - **Randomly generate the local administrator password and disable the account on all supported platforms (recommended)**: Use this setting to have the wizard randomly create a password for the local administrator account and disable the account when the operating system image is deployed.
  - **Enable the account and specify the local administrator password**: Use this setting to use a specific password for the local administrator account on all computers where the operating system image is deployed.

7. On the **Configure Network** page, specify the following settings, and then click **Next**.
   - **Join a workgroup**: Specify whether to add the destination computer to a workgroup.
   - **Join a domain**: Specify whether to add the destination computer to a domain. In **Domain**, specify the name of the domain.

     > **IMPORTANT**
     > You can browse to locate domains in the local forest, but you must specify the domain name for a remote forest.

     You can also specify an organizational unit (OU). This is an optional setting that specifies the LDAP X.500-distinguished name of the OU in which to create the computer account if it does not already exist.
   - **Account**: Specify the user name and password for the account that has permissions to join the specified domain. For example: *domain\user* or *%variable%*.
8. On the **Install Configuration Manager** page, specify the Configuration Manager client package to install on the destination computer, and then click **Next**.
9. On the **Install Applications** page, specify the applications to install on the destination computer, and then click **Next**. If you specify multiple applications, you can also specify that the task sequence continues if the installation of a specific application fails.
10. Complete the wizard.

Create a VHD

After you create a task sequence for the VHD, use the Create Virtual Hard Disk Wizard to create the VHD.

> **IMPORTANT**
> Before you run this procedure, verify that you meet the prerequisites listed at the start of this topic.

Use the following procedure to create a VHD.

To create a VHD

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Virtual Hard Disks**.
3. On the **Home** tab, in the **Create** group, click **Create Virtual Hard Disk** to start the Create Virtual Hard Disk Wizard.

   > **NOTE**
   > Hyper-V must be installed on the computer running the Configuration Manager console from which you manage VHDs or the **Create Virtual Hard Disk** option is not enabled. For more information about Hyper-V requirements, see Hyper-V Installation Prerequisites.

   > **TIP**
   > To organize your VHDs, create a new folder or select an existing folder under the **Virtual Hard Disks** node, and then click **Create Virtual Hard Disk** from the folder.

4. On the **General** page, specify the following settings, and then click **Next**.
   - **Name**: Specify a unique name for the VHD.
   - **Version**: Specify a version number for the VHD. This is an optional setting.
   - **Comment**: Specify a description for the VHD.
   - **Path**: Specify the path and file name for where the wizard will create the VHD file.

     You must enter a valid network path in the UNC format. For example: **\\servername\\.vhd**.

     > **WARNING**
     > Configuration Manager must have **Write** access permission to the specified path to create the VHD. When Configuration Manager fails to access the path, it logs the associated error in the distmgr.log file on the site server.

5. On the **Task Sequence** page, specify the task sequence that you specified in the previous section, and then click **Next**.
6. On the **Distribution Points** page, select one or more distribution points that contain the content required by the task sequence, and then click **Next**.
7. On the **Customization** page, click **Next**. The process to create the VHD ignores any settings that you specify on this page.
8. Verify the settings and then click **Next**. The wizard creates the VHD.

TIP

The time to complete the process to create the VHD can vary. While the wizard works through this process, you can monitor the following log files to track the progress. By default, the logs are located on the computer running the Configuration Manager console at *%ProgramFiles(x86)*%\Microsoft Configuration Manager\AdminConsole\AdminUILog.

- **CreateTSMedia.log**: The wizard writes information to this log while it creates the task sequence media. Review this log file to track the progress of the wizard when it creates the standalone media.
  - **DeployToVHD.log**: The wizard writes information to this log while it goes through the process to create the VHD. Review this log file to track the progress of the wizard for all steps after it creates the standalone media.

Also, when the operating system installation starts, you can open Hyper-V Manager (if you installed the Hyper-V management tools on the computer) and connect to the temporary virtual machine created by the wizard to see the task sequence running. From the virtual machine, you can monitor the smsts.log file to track the progress of the task sequence. When there are issues with completing a task sequence step, you can use this log file to help you troubleshoot the issue. The smsts.log file is in x: \windows\temp\smstslog\smsts.log before the hard disk if formatted and in c:_SMSTaskSequence\Logs\Smstslog\ after it is formatted. After the task sequence steps complete, the virtual machine is shut down after 5 minutes (by default) and deleted.

After Configuration Manager creates the VHD, it is located in the **Virtual Hard Drives** node in the Configuration Manager console under the **Operating System Deployment** node in the **Software Library** workspace.

NOTE

Configuration Manager retrieves the size of the VHD by connecting to the source location of the VHD. If Configuration Manager cannot access the VHD file, **0** is displayed in the **Size (KB)** column for the VHD.

Steps to Modify an Existing VHD

To modify a VHD, you must create a task sequence with the steps required to modify the VHD. Then, select the task sequence in the Modify Virtual Hard Drive Wizard. The wizard attaches the VHD to the virtual machine, runs the task sequence in the VHD, and then updates the VHD file. The following sections provide the steps to modify the VHD.

Create a Task Sequence to Modify the VHD

To modify an existing VHD, you must first create a task sequence. Choose only the steps that are required to modify the task sequence. For example, if you want to add an application to the VHD, create a custom task sequence, and then add only the Install Application step.

Use the following procedure to create the task sequence to modify the VHD.

To create a custom task sequence to modify the VHD

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, select **Create a new custom task sequence**, and then click **Next**.
5. On the **Task Sequence Information** page, specify the following settings, and then click **Next**.
   - **Task sequence name**: Specify a name that identifies the task sequence.

- **Description**: Specify a description of the task sequence.
- **Boot image**: Specify the boot image that installs the operating system on the destination computer. For more information, see Manage boot images.

6. Complete the wizard.

Use the following procedure to add task sequence steps to the custom task sequence.

To add task sequence steps to the custom task sequence

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, click **Task Sequences**, and then select the custom task sequence that you created in the previous procedure.
3. On the **Home** tab, in the **Task Sequence** group, click **Edit** to start the task sequence editor.
4. Add the task sequence steps to use to modify the VHD.
5. Click **OK** to exit the task sequence editor.

Modify a VHD

After you create a task sequence for the VHD, use the Modify Virtual Hard Disk Wizard to modify the VHD.

Use the following procedure to modify a VHD.

To modify a VHD

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, click **Virtual Hard Disks**, and then select the VHD to modify.
3. On the **Home** tab, in the **Virtual Hard Disk** group, click **Modify Virtual Hard Disk** to start the Modify Virtual Hard Disk Wizard.

   > **NOTE**
   > Hyper-V must be installed on the computer running the Configuration Manager console from which you manage VHDs or the **Modify Virtual Hard Disk** option is not enabled. For more information about Hyper-V requirements, see Hyper-V Installation Prerequisites.

4. On the **General** page, confirm the following settings, and then click **Next**.
   - **Name**: Specifies the unique name for the VHD.
   - **Version**: Specifies the version number for the VHD. This is an optional setting.
   - **Comment**: Specifies the description for the VHD.
   - **Path**: Specifies the path and file name for where the VHD file is located. You cannot modify this setting.

     > **WARNING**
     > Configuration Manager must have **Write** access permission to the specified path to create the VHD. When Configuration Manager fails to access the path, it logs the associated error in the distmgr.log file on the site server.

5. On the **Task Sequence** page, specify the custom task sequence that you created in the previous section, and

then click **Next**.

6. On the **Distribution Points** page, select one or more distribution points that contain the content required by the task sequence, and then click **Next**.

7. On the **Customization** page, click **Next**. The process to modify the VHD ignores any settings that you specify on this page.

8. Verify the settings and then click **Next**. The wizard creates the modified VHD.

> **TIP**
>
> The time to complete the process to modify the VHD can vary. While the wizard works through this process, you can monitor the following log files to track the progress. By default, the logs are located on the computer running the Configuration Manager console at *%ProgramFiles(x86)*%\Microsoft Configuration Manager\AdminConsole\AdminUILog.
>
> - **CreateTSMedia.log**: The wizard writes information to this log while it creates the task sequence media. Review this log file to track the progress of the wizard when it creates the standalone media.
>   - **DeployToVHD.log**: The wizard writes information to this log while it goes through the process to modify the VHD. Review this log file to track the progress of the wizard for all steps after it creates the standalone media.
>
> Also, you can open Hyper-V Manager (if you installed the Hyper-V management tools on the computer) and connect to the temporary virtual machine created by the wizard to see the task sequence running. From the virtual machine, you can monitor the smsts.log file to track the progress of the task sequence. When there are issues with completing a task sequence step, you can use this log file to help you troubleshoot the issue. The smsts.log file is in x:\windows\temp\smstslog\smsts.log before the hard disk if formatted and in c:_SMSTaskSequence\Logs\Smstslog\ after it is formatted. After the task sequence steps complete, the virtual machine is shut down after 5 minutes (by default) and deleted.

Apply Software Updates to a VHD

Periodically, new software updates are released that are applicable to the operating system in your VHD. You can apply applicable software updates to a VHD on a specified schedule. On the schedule that you specify, Configuration Manager applies the software updates that you select to the VHD.

Information about the VHD is stored in the site database, including the software updates that were applied at the time you created the VHD. Software updates that have been applied to the VHD since it was initially created are also stored in the site database. When you start the wizard to apply software updates to the VHD, the wizard retrieves a list of applicable software updates that have not yet been applied to the VHD for you to select.

You can select the **Continue on error** setting for Configuration Manager to continue to apply software updates even when there is an error applying one or more of the software updates that you selected.

> **NOTE**
>
> The software updates are copied from the content library on the site server.

Use the following procedure to apply software updates to VHD.

To apply software updates to a VHD

1. In the Configuration Manager console, click **Software Library**.

2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Virtual Hard Disks**.

3. Select the VHD to apply software updates.

4. On the **Home** tab, in the **Virtual Hard Disk** group, click **Schedule Updates** to start the wizard.
5. On the **Choose Updates** page, select the software updates to apply to the VHD, and then click **Next**.
6. On the **Set Schedule** page, specify the following settings, and then click **Next**.
    a. **Schedule**: Specify the schedule for when the software updates are applied to the VHD.
    b. **Continue on error**: Select this option to continue to apply software updates to the image even when there is an error.
7. On the **Summary** page, verify the information, and then click **Next**.
8. On the **Completion** page, verify that the software updates were successfully applied to the operating system image.

Import the VHD to System Center Virtual Machine Manager

System Center VMM is a management solution for the virtualized datacenter, enabling you to configure and manage your virtualization host, networking, and storage resources in order to create and deploy virtual machines and services to private clouds that you have created. After you create a VHD in Configuration Manager, you can import and manage your VHD by using VMM.

> **TIP**
> Before you upload a VHD to VMM, verify that the VMM console successfully connects to the VMM management server.

Use the following procedure to import a VHD to VMM.

To import a VHD to VMM

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Virtual Hard Disks**.
3. On the **Home** tab, in the **Virtual Hard Disk** group, click **Upload to Virtual Machine Manager** to start the Upload to Virtual Machine Manager Wizard.
4. On the **General** page, configure the following settings, and then click **Next**.
    - **VMM server name**: Specify the FQDN of the computer on which the VMM management server is installed. The wizard connects to the VMM management server to download the library shares for the server.
    - **VMM library share**: Specify the VMM library share from the drop-down list.
    - **Use unencrypted transfer**: Select this setting to transfer the VHD file to the VMM management server without the use of encryption.
5. On the Summary page, verify the settings, and then complete the wizard. The time it takes to upload the VHD can vary depending on the size of the VHD file and the network bandwidth to the VMM management server.

11/23/2016 • 1 min to read • Edit Online

Create a custom task sequence with System Center Configuration Manager

11/23/2016 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

When you create a custom task sequence in System Center Configuration Manager, it contains no task sequence steps. After you create the task sequence, you must edit it and add the task sequence steps you need.

Create a custom task sequence

Use the following procedure to create a custom task sequence.

To create a custom task sequence

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence** to start the Create Task Sequence Wizard.
4. On the **Create a New Task Sequence** page, select **Create a new custom task sequence**.
5. On the **Task Sequence Information** page, specify a name for the task sequence, a description of the task sequence, and an optional boot image for the task sequence to use, and then complete the wizard.

After you complete the Create Task Sequence Wizard, Configuration Manager adds the custom task sequence to the **Task Sequences** node. You can now edit this task sequence to add task sequence steps to it.

For a list of available task sequence steps, see Task sequence steps.

For more information about how to edit a task sequence, see Edit a task sequence.

Most often you will use task sequences to automate tasks for operating system deployment, but you can create a custom task sequence to automate a variety of tasks. For more information, see Create a task sequence for non-operating system deployments.

Next steps

Deploy the task sequence

Create a task sequence for non-operating system deployments with System Center Configuration Manager

1/19/2017 • 1 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Task sequences in System Center Configuration Manager are used to automate a variety of tasks within your environment. These tasks are primarily designed and tested for deploying operating systems. Configuration Manager has many other features that should be the primary technology that you use for scenarios such as application installation, software updates installation, setting configuration, or custom automation. There are other Microsoft System Center automation technologies, such as Orchestrator and Service Management Automation that you should also consider.

The power of task sequences lies in their flexibility and how you can use them to configure client settings, distribute software, update drivers, edit user states, and perform other tasks independent of operating system deployment. You can create a custom task sequence to add any number of tasks. You can create a custom task sequence to add any number of tasks. The use of custom task sequences for non-operating system deployment is supported in Configuration Manager. However, if a task sequence results in unwanted or inconsistent results, look at ways to simplify the operation. You can accomplish this by using simpler steps, dividing the actions across multiple task sequences, or by taking a phased approach to creating and testing the task sequence.

The following steps are supported for use in a non-operating system deployment custom task sequence:

- Check Readiness
- Connect To Network Folder
- Download Package Content
- Install Application
- Install Package
- Install Software Updates
- Restart Computer
- Run Command Line
- Run PowerShell Script
- Set Dynamic Variables
- Set Task Sequence Variable

Next steps

Deploy the task sequence

Task sequence steps to manage BIOS to UEFI conversion

3/26/2017 • 4 min to read • Edit Online

Windows 10 provides many new security features that require UEFI-enabled devices. You might have modern Windows PCs that support UEFI, but are using legacy BIOS. Converting a device to UEFI has required you to go to each PC, repartition the hard disk, and reconfigure the firmware. By using task sequences in Configuration Manager, you can prepare a hard drive for BIOS to UEFI conversion, convert from BIOS to UEFI as part of the in-place upgrade process, and collect UEFI information as part of hardware inventory.

Hardware inventory collects UEFI information

Beginning in version 1702, a new hardware inventory class (**SMS_Firmware**) and property (**UEFI**) are available to help you determine whether a computer starts in UEFI mode. When a computer is started in UEFI mode, the **UEFI** property is set to **TRUE**. This is enabled in hardware inventory by default. For more information about hardware inventory, see How to configure hardware inventory.

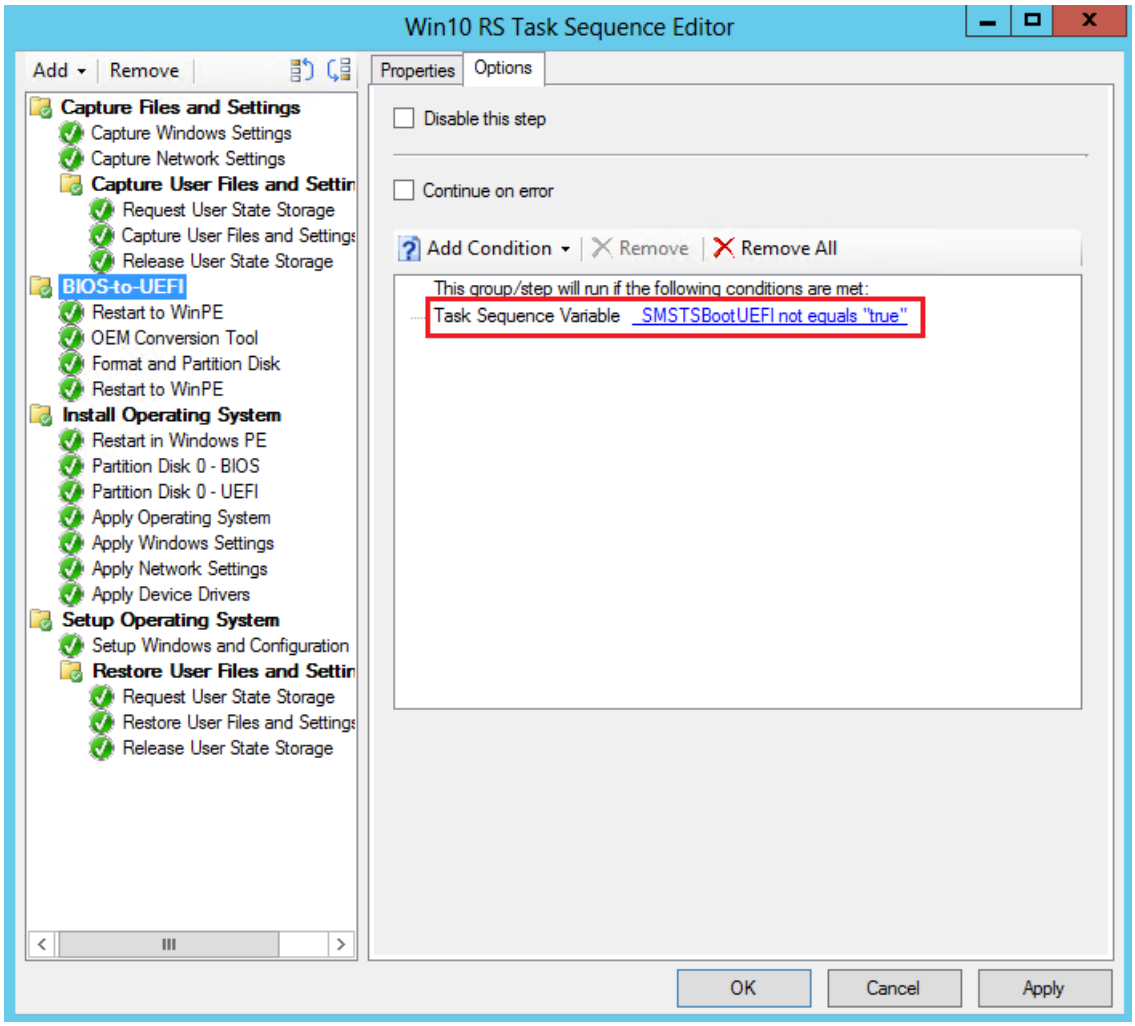
Create a custom task sequence to prepare the hard drive for BIOS to UEFI conversion

Starting in Configuration Manager version 1610, you can now customize an operating system deployment task sequence with a new variable, TSUEFIDrive, so that the **Restart Computer** step will prepare a FAT32 partition on the hard drive for transition to UEFI. The following procedure provides an example of how you can create task sequence steps to prepare the hard drive for the BIOS to UEFI conversion.

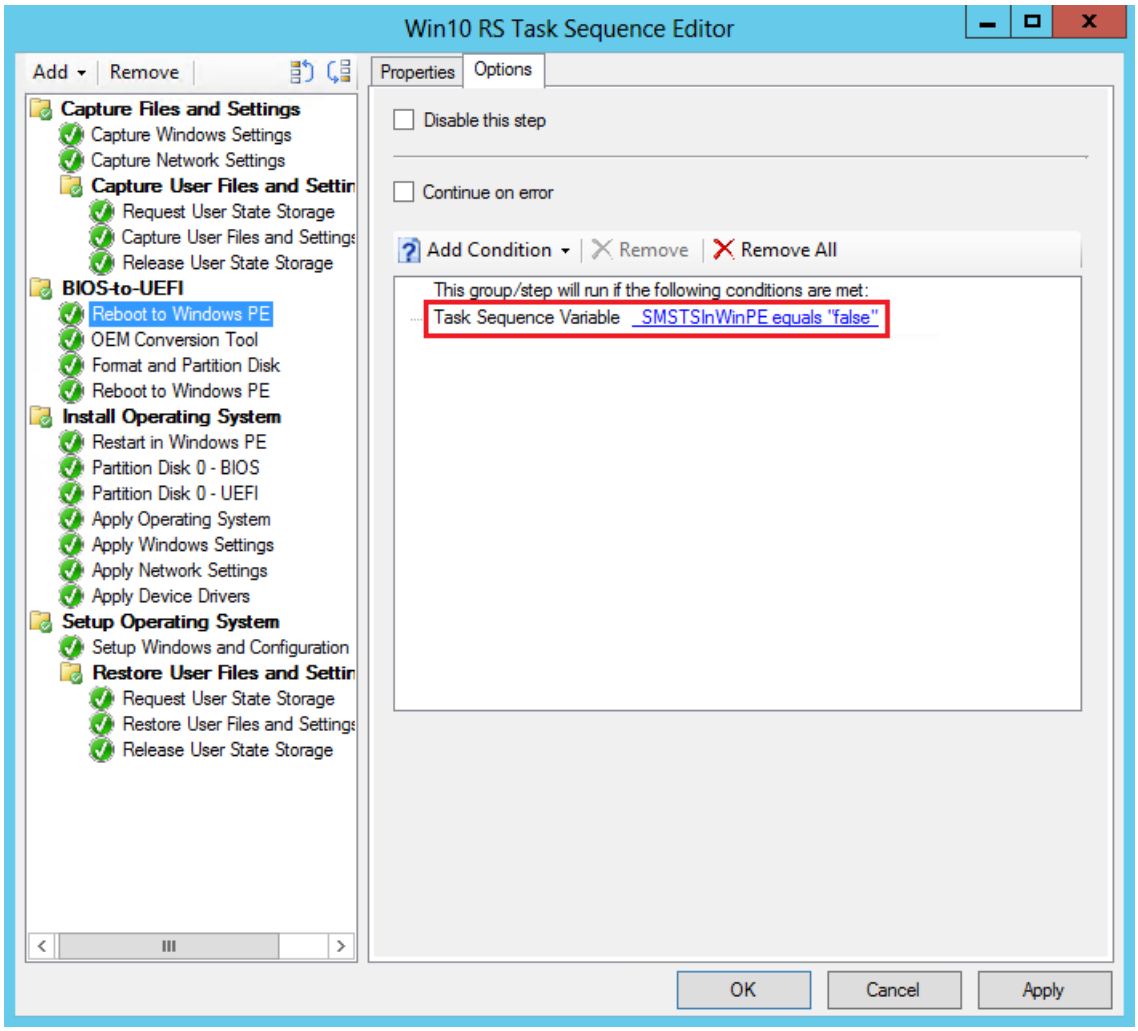
To prepare the FAT32 partition for the conversion to UEFI:

In an existing task sequence to install an operating system, you will add a new group with steps to do the BIOS to UEFI conversion.

1. Create a new task sequence group after the steps to capture files and settings, and before the steps to install the operating system. For example, create a group after the **Capture Files and Settings** group named **BIOS-to-UEFI**.
2. On the **Options** tab of the new group, add a new task sequence variable as a condition where **_SMSTSBootUEFI** is **not equal** to **true**. This prevents the steps in the group from running when a computer is already in UEFI mode.

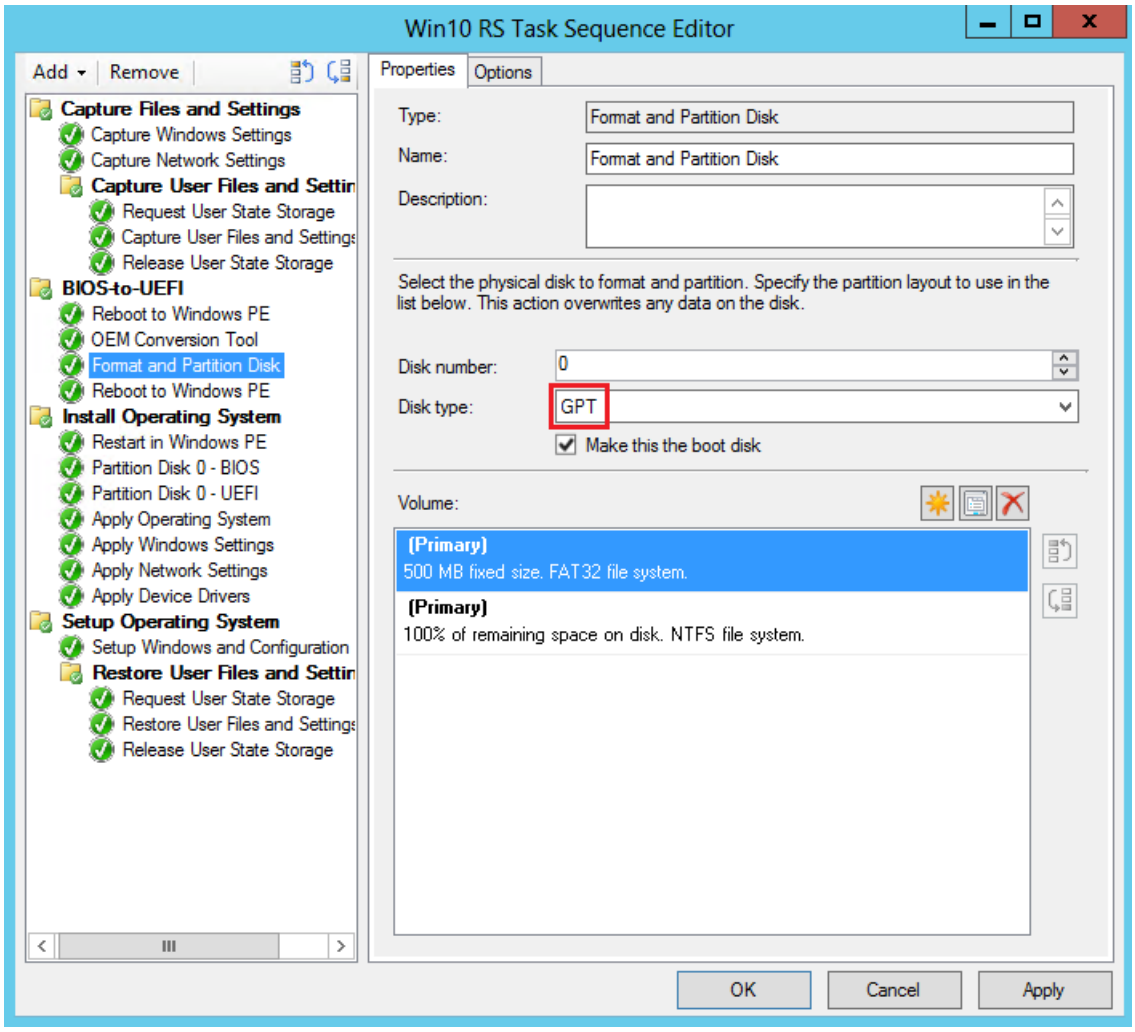


3. Under the new group, add the **Restart Computer** task sequence step. In **Specify what to run after restart**, select **The boot image assigned to this task sequence is selected** to start the computer in Windows PE.
4. On the **Options** tab, add a task sequence variable as a condition where **_SMSTSInWinPE equals false**. This prevents this step from running if the computer is already in Windows PE.

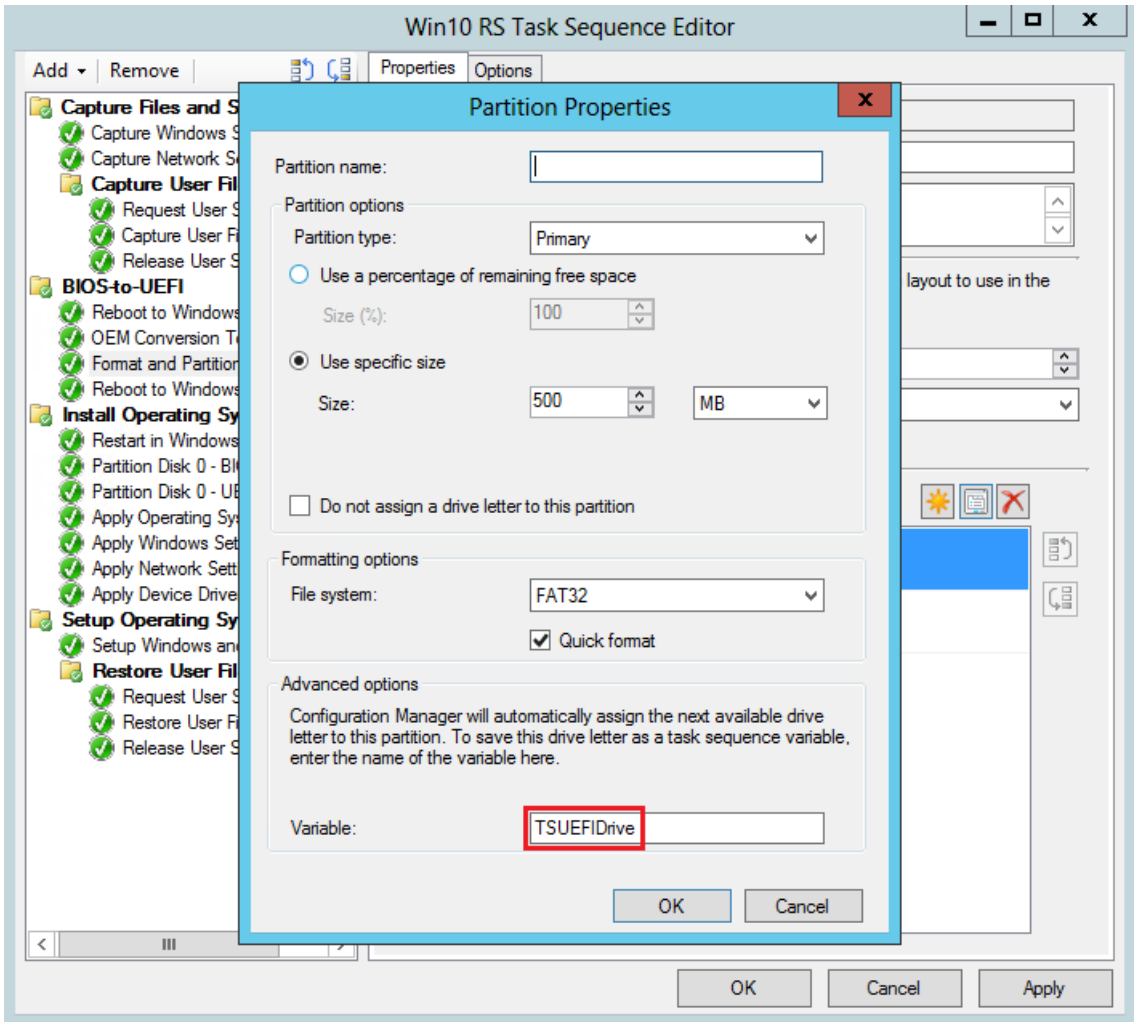


5. Add a step to start the OEM tool that will convert the firmware from BIOS to UEFI. This will typically be a **Run Command Line** task sequence step with a command line to start the OEM tool.
6. Add the Format and Partition Disk task sequence step that will partition and format the hard drive. In the step, do the following:

    a. Create the FAT32 partition that will be converted to UEFI before the operating system is installed. Choose **GPT** for **Disk type**.



b. Go to the properties for the FAT32 partition. Enter **TSUEFIDrive** in the **Variable** field. When the task sequence detects this variable, it will prepare for the UEFI transition before restarting the computer.



c. Create an NTFS partition that the task sequence engine uses to save its state and to store log files.

7. Add the **Restart Computer** task sequence step. In **Specify what to run after restart**, select **The boot image assigned to this task sequence is selected** to start the computer in Windows PE.

Convert from BIOS to UEFI during an in-place upgrade

Windows 10 Creators Update introduces a simple conversion tool that automates the process to repartition the hard disk for UEFI-enabled hardware and integrates the conversion tool into the Windows 7 to Windows 10 in-place upgrade process. When you combine this tool with your operating system upgrade task sequence and the OEM tool that converts the firmware from BIOS to UEFI, you can convert your computers from BIOS to UEFI during an in-place upgrade to the Windows 10 Creators Update.

Requirements:

- Windows 10 Creators Update
- Computers that support UEFI
- OEM tool that converts the computer's firmware from BIOS to UEFI

To convert from BIOS to UEFI during an in-place upgrade

1. Create an operating system upgrade task sequence that performs an in-place upgrade to Windows 10 Creators Update.
2. Edit the task sequence. In the **Post-Processing group**, add the following task sequence steps:

   a. From General, add a **Run Command Line** step. You will add the command line for the MBR2GPT tool that coverts a disk from MBR to GPT without modifying or deleting data from the disk. In Command line, type the following: **MBR2GPT /convert /disk:0 /AllowFullOS**.

NOTE

> You can also choose to run the MBR2GPT.EXE tool when in Windows PE instead of in the full operating system. You can do this by adding a step to restart the computer to WinPE before the step to run the MBR2GPT.EXE tool and removing the /AllowFullOS option from the command line. For details about the tool and available options, see MBR2GPT.EXE.

b. Add a step to start the OEM tool that will convert the firmware from BIOS to UEFI. This will typically be a Run Command Line task sequence step with a command line to start the OEM tool.
c. From General, add the **Restart Computer** step. For Specify what to run after restart, select **The currently installed default operating system**.

3. Deploy the task sequence.

Create task sequence media with System Center Configuration Manager

11/23/2016 • 4 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

You can use media to capture an operating system image from a reference computer or to deploy an operating system to a destination computer in your System Center Configuration Manager environment. The media that you create can be a CD, DVD set, or a USB flash drive.

Media is used mostly to deploy operating systems on destination computers that do not have a network connection or that have a low bandwidth connection to your Configuration Manager site. However, deployment media is also used to start an operating system deployment outside of an existing Windows operating system. This second use of deployment media is important for times when there is no operating system on the destination computer, the operating system is in a non-operable state, or the administrative user wants to repartition the hard disk on the destination computer.

Deployment media includes bootable media, stand-alone media, and prestaged media. The content of the deployment media varies, depending on what type of media that you use. For example, stand-alone media contains the task sequence that deploys the operating system while other types of media retrieve task sequences from the management point.

> **IMPORTANT**
> To create task sequence media, you must be an administrator on the computer from which you run the Configuration Manager console. If you are not an administrator, you will be prompted for administrator credentials when you start the Create Task Sequence Media wizard.

Capture media for operating system images

Capture media allows you to capture an operating system image from a reference computer. Capture media contains the boot image that starts the reference computer and the task sequence that captures the operating system image. For information about how to create capture media, see Create capture media with System Center Configuration Manager.

Bootable media operating system deployments

Bootable media contains only the boot image, optional prestart commands and their required files, and Configuration Manager binaries. When the destination computer starts, it connects to the network and retrieves the task sequence, the operating system image, and any other required content from the network. Because the task sequence is not on the media, you can change the task sequence or content without having to recreate the media.

> **IMPORTANT**
> The packages on bootable media are not encrypted. The administrative user must take the appropriate security measures, such as adding a password to the media, to ensure that the package contents are secured from unauthorized users.

For information about how to create bootable media, Create bootable media.

Prestaged media operating system deployments

Prestaged media allows you to prestage bootable media and an operating system image to a hard disk prior to the provisioning process. The prestaged media is a Windows Imaging Format (WIM) file that can be installed on a bare-metal computer by the manufacturer or at an enterprise staging center that is not connected to the Configuration Manager environment.

Prestaged media contains the boot image used to start the destination computer and the operating system image that is applied to the destination computer. You can also specify applications, packages, and driver packages to include as part of the prestaged media. The task sequence that deploys the operating system is not included in the media. When you deploy a task sequence that uses prestaged media, the client checks the local task sequence cache for valid content first, and if the content cannot be found or has been revised, the client downloads the content from the distribution point.

Prestaged media is applied to the hard drive of a new computer before the computer is sent to the end user. When the computer starts for the first time after the prestaged media has been applied, the computer starts Windows PE and connects to a management point to locate the task sequence that completes the operating system deployment process.

> **IMPORTANT**
> The packages on prestaged media are not encrypted. The administrative user must take the appropriate security measures, such as adding a password to the media, to ensure that the package contents are secured from unauthorized users.

For information about how to create prestaged media, see Create prestaged media.

Stand-alone media operating system deployments

Stand-alone media contains everything that is required to deploy the operating system. This includes the task sequence and any other required content. Because everything that is required to deploy the operating system is stored on the stand-alone media, the disk space required for stand-alone media is significantly larger than the disk space required for other types of media.

For information about how to create stand-alone media, see Create stand-alone media.

Media considerations when using site systems configured for HTTPS

When your management point and distribution points are configured to use HTTPS communication, you must create boot media and prestaged media at a primary site, not the central administration site. Also, consider the following to help you determine whether to configure the media as dynamic or site-based:

- To configure the media as dynamic media, all primary sites must have the root CA of the site from which you created the media. You can import the root CA to all primary sites in your hierarchy.
- When primary sites in your Configuration Manager hierarchy use different root CAs, you must use site-based media at each site.

Create stand-alone media with System Center Configuration Manager

3/26/2017 • 12 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Stand-alone media in Configuration Manager contains everything that is required to deploy the operating system on a computer without a connection to a Configuration Manager site or using the network. Use stand-alone media with the following operating system deployment scenarios:

- Refresh an existing computer with a new version of Windows
- Install a new version of Windows on a new computer (bare metal)
- Upgrade Windows to the latest version

Stand-alone media includes the task sequence that automates the steps to install the operating system and all other required content, including the boot image, operating system image, and device drivers. Because everything to deploy the operating system is stored on the stand-alone media, the disk space required for stand-alone media is significantly larger than the disk space required for other types of media. When you create stand-alone media on a central administration site, the client will retrieve its assigned site code from active directory. Stand-alone media created at child sites will automatically assign to the client the site code for that site.

Create stand-alone media

Before you create stand-alone media by using the Create Task Sequence Media Wizard, be sure that the following conditions are met:

Create a task sequence to deploy an operating system

As part of the stand-alone media, you must specify the task sequence to deploy an operating system. For the steps to create a new task sequence, see Create a task sequence to install an operating system in System Center Configuration Manager.

The following actions are not supported for stand-alone media:

- The Auto Apply Drivers step in the task sequence. Automatic application of device drivers from the driver catalog is not supported, but you can choose the Apply Driver Package step to make a specified set of drivers available to Windows Setup.
- The Download Package Content step in the task sequence. The management point information is not available on standalone media, therefore the step will fail trying to enumerate content locations.
- Installing software updates.
- Installing software before deploying the operating system.
- Task sequences for non-operating system deployments.
- Associating users with the destination computer to support user device affinity.
- Dynamic package installs via the Install Packages task.
- Dynamic application installs via the Install Application task.

If your task sequence to deploy an operating system includes the Install Package step and you create the stand-alone media at a central administration site, an error might occur. The central administration site does not have the necessary client configuration policies that are required to enable the software distribution agent during the execution of the task sequence. The following error might appear in the CreateTsMedia.log file:

"WMI method SMS_TaskSequencePackage.GetClientConfigPolicies failed (0x80041001)"

For stand-alone media that includes an **Install Package** step, you must create the stand-alone media at a primary site that has the software distribution agent enabled or add a Run Command Line step after the Setup Windows and ConfigMgr step and before the first **Install Package** step in the task sequence. The **Run Command Line** step runs a WMIC command to enable the software distribution agent before the first Install package step runs. You can use the following in your **Run Command Line** task sequence step:

```
WMIC /namespace:\\\root\ccm\policy\machine\requestedconfig path ccm_SoftwareDistributionClientConfig CREATE
ComponentName="Enable SWDist", Enabled="true", LockSettings="TRUE", PolicySource="local",
PolicyVersion="1.0", SiteSettingsKey="1" /NOINTERACTIVE
```

Distribute all content associated with the task sequence

You must distribute all content that is required by the task sequence the to at least one distribution point. This includes the boot image, operating system image, and other associated files. The wizard gathers the information from the distribution point when it creates the stand-alone media. You must have **Read** access rights to the content library on that distribution point. For details, see Distribute content referenced by a task sequence.

Prepare the removable USB drive

For a removable USB drive:

If you are going to use a removable USB drive, the USB drive must be connected to the computer where the wizard is run and the USB drive must be detectable by Windows as a removal device. The wizard writes directly to the USB drive when it creates the media. Stand-alone media uses a FAT32 file system. You cannot create stand-alone media on a USB flash drive whose content contains a file over 4 GB in size.

Create an output folder

For a CD/DVD set:

Before you run the Create Task Sequence Media Wizard to create media for a CD or DVD set, you must create a folder for the output files created by the wizard. Media that is created for a CD or DVD set is written as .iso files directly to the folder.

Use the following procedure to create stand-alone media for a removable USB drive or a CD/DVD set.

To create stand-alone media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, specify the following options, and then click **Next**.
   - Select **Stand-alone media**.
   - Optionally, if you want to allow the operating system to be deployed without requiring user input, select **Allow unattended operating system deployment**. When you select this option the user is not prompted for network configuration information or for optional task sequences. However, the user is still prompted for a password if the media is configured for password protection.
5. On the **Media Type** page, specify whether the media is a flash drive or a CD/DVD set, and then click

configure the following:

> **IMPORTANT**
> Stand-alone media uses a FAT32 file system. You cannot create stand-alone media on a USB flash drive whose content contains a file over 4 GB in size.

- If you select **USB flash drive**, specify the drive where you want to store the content.
- If you select **CD/DVD set**, specify the capacity of the media and the name and path of the output files. The wizard writes the output files to this location. For example: **\\servername\folder\outputfile.iso**

  If the capacity of the media is too small to store the entire content, multiple files are created and you must store the content on multiple CDs or DVDs. When multiple media is required, Configuration Manager adds a sequence number to the name of each output file that it creates. In addition, if you deploy an application along with the operating system and the application cannot fit on a single media, Configuration Manager stores the application across multiple media. When the stand-alone media is run, Configuration Manager prompts the user for the next media where the application is stored.

  > **IMPORTANT**
  > If you select an existing .iso image, the Task Sequence Media Wizard deletes that image from the drive or share as soon as you proceed to the next page of the wizard. The existing image is deleted, even if you then cancel the wizard.

Click **Next**.

6. On the **Security** page, choose from the following settings, and then click **Next**:

   - **Protect media with a password**: Enter a strong password to help protect the media. If you specify a password, the password is required to use the media.

     > **IMPORTANT**
     > On stand-alone media, only the task sequence steps and their variables are encrypted. The remaining content of the media is not encrypted, so do not include any sensitive information in task sequence scripts. Store and implement all sensitive information by using task sequence variables.

   - **Select date range for this stand-alone media to be valid** (starting in version 1702): Set optional start and expiration dates on the media. These settings are disabled by default. The dates are compared to the system time on the computer before the stand-alone media runs. When the system time is earlier than the start time or later than the expiration time, the stand-alone media is not started. These options are also available by using the New-CMStandaloneMedia PowerShell cmdlet.

7. On the **Stand-Alone CD/DVD** page, specify the task sequence that deploys the operating system, and then click **Next**. Choose **Detect associated application dependencies and add them to this media** to add content to the stand-alone media for application dependencies.

> **TIP**
> If you do not see expected application dependencies, deselect and then reselect the **Detect associated application dependencies and add them to this media** setting to refresh the list.

The wizard lets you select only those task sequences that are associated with a boot image.

8. On the **Select Application** page (available beginning in version 1702), specify application content to include as part of the media file, and then click **Next**.
9. On the **Select Package** page (available beginning in version 1702), specify the package content to include as part of the media file, and then click **Next**.
10. On the **Select Driver Package** page (available beginning in version 1702), specify the driver package content to include as part of the media file, and then click **Next**.
11. On the **Distribution Points** page, specify the distribution points that contain the content required by the task sequence, and then click **Next**.

    Configuration Manager will only display distribution points that have the content. You must distribute all of the content associated with the task sequence (boot image, operating system image, etc.) to at least one distribution point before you can continue. After you distribute the content, you can either restart the wizard or remove any distribution points that you already selected on this page, go to the previous page, and then back to the **Distribution Points** page to refresh the distribution point list. For more information about distributing content, see Distribute content referenced by a task sequence. For more information about distribution points and content management, see Manage content and content infrastructure for System Center Configuration Manager.

```
> [!NOTE]
>  You must have **Read** access rights to the content library on the distribution points.
```

12. On the **Customization** page, specify the following information, and then click **Next**.

    - Specify the variables that the task sequence uses to deploy the operating system.
    - Specify any prestart commands that you want to run before the task sequence. Prestart commands are a script or an executable that can interact with the user in Windows PE before the task sequence runs to install the operating system. For more information about prestart commands for media, see Prestart commands for task sequence media in System Center Configuration Manager.

      Optionally, select **Files for the prestart command** to include any required files for the prestart command.

      > **TIP**
      > During task sequence media creation, the task sequence writes the package ID and prestart command-line, including the value for any task sequence variables, to the CreateTSMedia.log log file on the computer that runs the Configuration Manager console. You can review this log file to verify the value for the task sequence variables.

13. Complete the wizard.

The stand-alone media files (.iso) are created in the destination folder. If you selected **Stand-Alone CD/DVD**, you can now copy the output files to a set of CDs or DVDs.

Example task sequence for stand-alone media

Use the following table as a guide as you create a task sequence to deploy an operating system using stand-alone media. The table will help you decide the general sequence for your task sequence steps and how to organize and structure those task sequence steps into logical groups. The task sequence that you create might vary from this sample and can contain more or fewer task sequence steps and groups.

> **NOTE**
> You must always use the Task Sequence Media Wizard to create stand-alone media.

TASK SEQUENCE GROUP OR STEP	DESCRIPTION
Capture File and Settings - **(New Task Sequence Group)**	Create a task sequence group. A task sequence group keeps similar task sequence steps together for better organization and error control.
Capture Windows Settings	Use this task sequence step to identify the Microsoft Windows settings that are captured from the existing operating system on the destination computer prior to reimaging. You can capture the computer name, user and organizational information, and the time zone settings.
Capture Network Settings	Use this task sequence step to capture network settings from the computer that receives the task sequence. You can capture the domain or workgroup membership of the computer and the network adapter setting information.
Capture User Files and Settings - **(New Task Sequence Sub-Group)**	Create a task sequence group within a task sequence group. This sub-group contains the steps needed to capture user state data from the existing operating system on the destination computer prior to reimaging. Similar to the initial group that you added, this sub-group keeps similar task sequence steps together for better organization and error control.
Set Local State Location	Use this task sequence step to specify a local location using the protected path task sequence variable. The user state is stored on a protected directory on the hard drive.
Capture User State	Use this task sequence step to capture the user files and settings you want to migrate to the new operating system.
Install Operating System - **(New Task Sequence Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to install the operating system.
Reboot to Windows PE or hard disk	Use this task sequence step to specify restart options for the computer that receives this task sequence. This step will display a message to the user indicating that the computer will be restarted so that the installation can continue. This step uses the read-only **_SMSTSInWinPE** task sequence variable. If the associated value equals **false** the task sequence step will continue.
Apply Operating System	Use this task sequence step to install the operating system image onto the destination computer. This step deletes all files on that volume (with the exception of Configuration Manager-specific control files) and then applies all volume images contained in the WIM file to the corresponding sequential disk volume. You can also specify a **sysprep** answer file to configure which disk partition to use for the installation.

TASK SEQUENCE GROUP OR STEP	DESCRIPTION
Apply Windows Settings	Use this task sequence step to configure the Windows settings configuration information for the destination computer. The windows settings you can apply are user and organizational information, product or license key information, time zone, and the local administrator password.
Apply Network Settings	Use this task sequence step to specify the network or workgroup configuration information for the destination computer. You can also specify if the computer uses a DHCP server or you can statically assign the IP address information.
Apply Driver Package	Use this task sequence step to make all device drivers in a driver package available for use by Windows setup. All necessary device drivers must be contained on the stand-alone media.
Setup Operating System - **(New Task Sequence Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to install the Configuration Manager client.
Setup Windows and ConfigMgr	Use this task sequence step to install the Configuration Manager client software. Configuration Manager installs and registers the Configuration Manager client GUID. You can assign the necessary installation parameters in the **Installation properties** window.
Restore User Files and Settings - **(New Task Sequence Group)**	Create another task sequence sub-group. This sub-group contains the steps needed to restore the user state.
Restore User State	Use this task sequence step to initiate the User State Migration Tool (USMT) to restore the user state and settings that were captured from the Capture User State Action to the destination computer.

Create prestaged media with System Center Configuration Manager

11/23/2016 • 8 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Prestaged media in System Center Configuration Manager is a Windows Imaging Format (WIM) file that can be installed on a bare-metal computer by the manufacturer or at an enterprise staging center that is not connected to the Configuration Manager environment.
Prestaged media contains the boot image used to start the destination computer and the operating system image that is applied to the destination computer. You can also specify applications, packages, and driver packages to include as part of the prestaged media. The task sequence that deploys the operating system is not included in the media. Prestaged media is applied to the hard drive of a new computer before the computer is sent to the end user. Use prestaged media for the following operating system deployment scenarios:

- Create an image for an OEM in factory or a local depot
- Install a new version of Windows on a new computer (bare metal)
- Deploy Windows to Go

When the computer starts for the first time after the prestaged media has been applied, the computer boots to Windows PE and connects to a management point to locate the task sequence that completes the operating system deployment process. You can specify applications, packages, and driver packages to include as part of the prestaged media. When you deploy a task sequence that uses prestaged media, the wizard checks the local task sequence cache for valid content first, and if the content cannot be found or has been revised, the wizard downloads the content from the distribution point.

How to Create Prestaged Media

Before you create prestaged media by using the Create Task Sequence Media Wizard, be sure that all the following conditions are met:

TASK	DESCRIPTION
Boot image	Consider the following about the boot image that you will use in the task sequence to deploy the operating system: - The architecture of the boot image must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image. - Ensure that the boot image contains the network and mass storage drivers that are required to provision the destination computer.

TASK	DESCRIPTION
Create a task sequence to deploy an operating system	As part of the prestaged media, you must specify the task sequence to deploy the operating system. - For the steps to create a new task sequence, see Create a task sequence to install an operating system. - For more information about task sequences, see Manage task sequences to automate tasks.
Distribute all content associated with the task sequence	You must distribute to at least one distribution point all content that is required by the task sequence. This includes the boot image, operating system image, and other associated files. The wizard gathers the information from the distribution point when it creates the stand-alone media. You must have **Read** access rights to the content library on that distribution point. For details, see About the content library.
Hard drive on the destination computer	The hard drive of the destination computer must be formatted before the pre-staged media is staged onto the hard drive of the computer. If the hard drive is not formatted when the media is applied, the task sequence that deploys the operating system will fail when it attempts to start the destination computer.

> **NOTE**
> The Create Task Sequence Media Wizard sets the following task sequence variable condition on the media: **_SMSTSMedia = OEMMedia**. You can use this condition in your task sequence.

Use the following procedure to create prestaged media.

To create prestaged media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, specify the following information, and then click **Next**.
   - Select **Prestaged media**.
   - Optionally, if you want to allow the operating system to be deployed without requiring user input, select **Allow unattended operating system deployment**. When you select this option the user is not prompted for network configuration information or for optional task sequences. However, the user is still prompted for a password if the media is configured for password protection.
5. On the **Media Management** page, specify the following information, and then click **Next**.
   - Select **Dynamic media** if you want to allow a management point to redirect the media to another management point, based on the client location in the site boundaries.
   - Select **Site-based media** if you want the media to contact only the specified management point.
6. On the **Media Properties** page, specify the following information, and then click **Next**.
   - **Created by**: Specify who created the media.

- **Version**: Specify the version number of the media.
- **Comment**: Specify a unique description of what the media is used for.
- **Media file**: Specify the name and path of the output files. The wizard writes the output files to this location. For example: **\\servername\folder\outputfile.wim**

7. On the **Security** page, specify the following information, and then click **Next**.

   - Select the **Enable unknown computer support** check box to allow the media to deploy an operating system to a computer that is not managed by Configuration Manager. There is no record of these computers in the Configuration Manager database. For more information, see Prepare for unknown computer deployments.
   - Select the **Protect the media with a password** check box and enter a strong password to help protect the media from unauthorized access. When you specify a password, the user must provide that password to use the prestaged media.

     > **IMPORTANT**
     > As a security best practice, always assign a password to help protect the prestaged media.

   - For HTTP communications, select **Create self-signed media certificate**, and then specify the start and expiration date for the certificate.
   - For HTTPS communications, select **Import PKI certificate**, and then specify the certificate to import and its password.

     For more information about this client certificate that is used for boot images, see PKI certificate requirements.

   - **User Device Affinity**: To support user-centric management in Configuration Manager, specify how you want the media to associate users with the destination computer. For more information about how operating system deployment supports user device affinity, see Associate users with a destination computer.

     - Specify **Allow user device affinity with auto-approval** if you want the media to automatically associate users with the destination computer. This functionality is based on the actions of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and destination computer when it deploys the operating system to the destination computer.
     - Specify **Allow user device affinity pending administrator approval** if you want the media to associate users with the destination computer after approval is granted. This functionality is based on the scope of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and the destination computer, but waits for approval from an administrative user before the operating system is deployed.
     - Specify **Do not allow user device affinity** if you do not want the media to associate users with the destination computer. In this scenario, the task sequence does not associate users with the destination computer when it deploys the operating system.

8. On the **Task Sequence** page, specify the task sequence that will run on the destination computer. The content referenced by the task sequence is displayed in **This task sequence references the following content**. Verify that the content, and then click **Next**.

9. On the **Boot image** page, specify the following information, and then click **Next**.

   > **IMPORTANT**
   > The architecture of the boot image that is distributed must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image.

   - In the **Boot image** box, specify the boot image to start the destination computer. For more information, see Manage boot images.
   - In the **Distribution point** box, specify the distribution point where the boot image resides. The wizard retrieves the boot image from the distribution point and writes it to the media.

     > **NOTE**
     > You must have **Read** access rights to the content library on the distribution point. For more information, see About the content library.

   - If you selected **Site-based media** on the **Media Management** page of the wizard, in the **Management point** box, specify a management point from a primary site.
   - If selected **Dynamic media** on the **Media Management** page of the wizard, in the **Associated management points** box, specify the primary site management points to use and a priority order for the initial communications.

10. On the **Images** page, specify the following information, and then click **Next**.

    - In the **Image package** box, specify the operating system image. For more information, see Manage operating system images.
    - If the package contains multiple operating system images, in the **Image index** box, specify the image to deploy.
    - In the **Distribution point** box, specify the distribution point where the operating system image package resides. The wizard retrieves the operating system image from the distribution point and writes it to the media.

11. On the **Customization** page, specify the following information, and then click **Next**.

    - Specify the variables that the task sequence uses to deploy the operating system.
    - Specify any prestart commands that you want to run before the task sequence runs. Prestart commands are a script or an executable that can interact with the user in Windows PE before the task sequence runs to install the operating system. For more information about prestart commands for media, see the Prestart commands for task sequence media.

      > **TIP**
      > During task sequence media creation, the task sequence writes the package ID and prestart command-line, including the value for any task sequence variables, to the CreateTSMedia.log log file on the computer that runs the Configuration Manager console. You can review this log file to verify the value for the task sequence variables.

12. Complete the wizard.

Next steps

Scenarios to deploy enterprise operating systems

Create bootable media with System Center Configuration Manager

1/23/2017 • 9 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Bootable media in Configuration Manager contains the boot image, optional prestart commands and associated files, and Configuration Manager files. Use prestaged media for the following operating system deployment scenarios:

- Install a new version of Windows on a new computer (bare metal)
- Replace an existing computer and transfer settings

Create bootable media

When you boot to the bootable media, the destination computer starts, connects to the network and retrieves the specified task sequence, operating system image, and any other required content from the network. Because the task sequence is not on the media, you can change the task sequence or content without having to recreate the media. The packages on bootable media are not encrypted. You must take the appropriate security measures, such as adding a password to the media, to ensure that the package contents are secured from unauthorized users.

Before you create bootable media by using the Create Task Sequence Media Wizard, be sure that all the following conditions are met:

TASK	DESCRIPTION
Boot image	Consider the following about the boot image that you will use in the task sequence to deploy the operating system: - The architecture of the boot image must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image. - Ensure that the boot image contains the network and mass storage drivers that are required to provision the destination computer.
Create a task sequence to deploy an operating system	As part of the bootable media, you must specify the task sequence to deploy the operating system. For the steps to create a new task sequence, see Create a task sequence to install an operating system.
Distribute all content associated with the task sequence	You must distribute to at least one distribution point all content that is required by the task sequence. This includes the boot image and other associated prestart files. The wizard gathers the information from the distribution point when it creates the bootable media. You must have **Read** access rights to the content library on that distribution point. For details, see About the content library.

TASK	DESCRIPTION
Prepare the removable USB drive	For a removable USB drive: If you are going to use a removable USB drive, the USB drive must be connected to the computer where the wizard is run and the USB drive must be detectable by Windows as a removal device. The wizard writes directly to the USB drive when it creates the media. Stand-alone media uses a FAT32 file system. You cannot create stand-alone media on a USB flash drive whose content contains a file over 4 GB in size.
Create an output folder	For a CD/DVD set: Before you run the Create Task Sequence Media Wizard to create media for a CD or DVD set, you must create a folder for the output files created by the wizard. Media that is created for a CD or DVD set is written as .iso files directly to the folder.

Use the following procedure to create bootable media.

To create bootable media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, specify the following options, and then click **Next**.
   - Select **Bootable media**.
   - Optionally, if you want to only allow the operating system to be deployed without requiring user input, select **Allow unattended operating system deployment**.

     > **IMPORTANT**
     > When you select this option, the user is not prompted for network configuration information or for optional task sequences. However, the user is still prompted for a password if the media is configured for password protection.

5. On the **Media Management** page, specify one of the following options, and then click **Next**.
   - Select **Dynamic media** if you want to allow a management point to redirect the media to another management point, based on the client location in the site boundaries.
   - Select **Site-based media** if you want the media to contact only the specified management point.
6. On the **Media Type** page, specify whether the media is a flash drive or a CD/DVD set, and then click configure the following:

   > **IMPORTANT**
   > Stand-alone media uses a FAT32 file system. You cannot create stand-alone media on a USB flash drive whose content contains a file over 4 GB in size.

- If you select **USB flash drive**, specify the drive where you want to store the content.
- If you select **CD/DVD set**, specify the capacity of the media and the name and path of the output files. The wizard writes the output files to this location. For example: **\\servername\folder\outputfile.iso**

  If the capacity of the media is too small to store the entire content, multiple files are created and you must store the content on multiple CDs or DVDs. When multiple media is required, Configuration Manager adds a sequence number to the name of each output file that it creates. In addition, if you deploy an application along with the operating system and the application cannot fit on a single media, Configuration Manager stores the application across multiple media. When the stand-alone media is run, Configuration Manager prompts the user for the next media where the application is stored.

  > **IMPORTANT**
  > If you select an existing .iso image, the Task Sequence Media Wizard deletes that image from the drive or share as soon as you proceed to the next page of the wizard. The existing image is deleted, even if you then cancel the wizard.

Click **Next**.

7. On the **Security** page, specify the following options, and then click **Next**.

- Select the **Enable unknown computer support** check box to allow the media to deploy an operating system to a computer that is not managed by Configuration Manager. There is no record of these computers in the Configuration Manager database.

  Unknown computers include the following:

  - A computer where the Configuration Manager client is not installed
  - A computer that is not imported into Configuration Manager
  - A computer that is not discovered by Configuration Manager
- Select the **Protect the media with a password** check box and enter a strong password to help protect the media from unauthorized access. When you specify a password, the user must provide that password to use the bootable media.

  > **IMPORTANT**
  > As a security best practice, always assign a password to help protect the bootable media.

- For HTTP communications, select **Create self-signed media certificate**, and then specify the start and expiration date for the certificate.
- For HTTPS communications, select **Import PKI certificate**, and then specify the certificate to import and its password.

  For more information about this client certificate that is used for boot images, see PKI certificate requirements.
- **User Device Affinity**: To support user-centric management in Configuration Manager, specify how you want the media to associate users with the destination computer. For more information about how operating system deployment supports user device affinity, see Associate users with a destination computer.

- Specify **Allow user device affinity with auto-approval** if you want the media to automatically associate users with the destination computer. This functionality is based on the actions of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and destination computer when it deploys the operating system to the destination computer.
- Specify **Allow user device affinity pending administrator approval** if you want the media to associate users with the destination computer after approval is granted. This functionality is based on the scope of the task sequence that deploys the operating system. In this scenario, the task sequence creates a relationship between the specified users and the destination computer, but waits for approval from an administrative user before the operating system is deployed.
- Specify **Do not allow user device affinity** if you do not want the media to associate users with the destination computer. In this scenario, the task sequence does not associate users with the destination computer when it deploys the operating system.

8. On the **Boot image** page, specify the following options, and then click **Next**.

> **IMPORTANT**
> The architecture of the boot image that is distributed must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image.

- In the **Boot image** box, specify the boot image to start the destination computer.
- In the **Distribution point** box, specify the distribution point where the boot image resides. The wizard retrieves the boot image from the distribution point and writes it to the media.

  > **NOTE**
  > You must have **Read** access rights to the content library on the distribution point.

- If you create site-based bootable media on the **Media Management** page of the wizard, specify a management point from a primary site in the **Management point** box.
- If you create dynamic bootable media on the **Media Management** page of the wizard, specify the primary site management points to use, and a priority order for the initial communications in **Associated management points**.

9. On the **Customization** page, specify the following options, and then click **Next**.

- Specify the variables that the task sequence uses to deploy the operating system.
- Specify any prestart commands that you want to run before the task sequence runs. Prestart commands are a script or an executable that can interact with the user in Windows PE before the task sequence runs to install the operating system. For more information, see Prestart commands for task sequence media.

> **TIP**
> During task sequence media creation, the task sequence writes the package ID and prestart command-line, including the value for any task sequence variables, to the CreateTSMedia.log log file on the computer that runs the Configuration Manager console. You can review this log file to verify the value for the task sequence variables.

Optionally, select the **Files for the prestart command** check box to include any required files for the prestart command.

10. Complete the wizard.

Create bootable media on a USB drive from a network share

The information in this section helps you to create bootable media on a USB flash drive when the flash drive is not connected to the computer running the Configuration Manager console. To create the bootable media on the USB drive, you can create task sequence boot media, mount the ISO, and transfer the files from the ISO to the USB drive.

1. Create the task sequence boot media. On the **Media type** page, select **CD/DVD set**. The wizard writes the output files to the location that you specify. For example: **\\servername\folder\outputfile.iso**.
2. Prepare the removable USB drive. The drive must be formatted, empty, and bootable.
3. Mount the ISO from the share location and transfer the files from the ISO to the USB drive.

Next steps

Use bootable media to deploy Windows over the network

Create capture media with System Center Configuration Manager

1/23/2017 • 3 min to read • Edit Online

Applies to: System Center Configuration Manager (Current Branch)

Capture media in Configuration Manager allows you to capture an operating system image from a reference computer. Use capture media for the following scenario:

- Create a task sequence to capture an operating systems

How to Create Capture Media

Use capture media to capture an operating system image from a reference computer. Capture media contains the boot image that starts the reference computer and the task sequence that captures the operating system image.

You create capture media by using the Create Task Sequence Media Wizard. Before you run the wizard, be sure that all the following conditions are met:

TASK	DESCRIPTION
Boot image	Consider the following about the boot image that you will use in the task sequence to capture the operating system: - The architecture of the boot image must be appropriate for the architecture of the destination computer. For example, an x64 destination computer can boot and run an x86 or x64 boot image. However, an x86 destination computer can boot and run only an x86 boot image. - Ensure that the boot image contains the network and mass storage drivers that are required to provision the destination computer.
Distribute all content associated with the task sequence	You must distribute to at least one distribution point all content that is required by the task sequence. This includes the boot image, operating system image, and other associated files. The wizard gathers the information from the distribution point when it creates the stand-alone media. You must have **Read** access rights to the content library on that distribution point. For more information, see Distribute content.
Prepare the removable USB drive	For a removable USB drive: If you are going to use a removable USB drive, the USB drive must be connected to the computer where the wizard is run and the USB drive must be detectable by Windows as a removal device. The wizard writes directly to the USB drive when it creates the media.

TASK	DESCRIPTION
Create an output folder	For a CD/DVD set: Before you run the Create Task Sequence Media Wizard to create media for a CD or DVD set, you must create a folder for the output files created by the wizard. Media that is created for a CD or DVD set is written as .iso files directly to the folder.

Use the following procedure to create capture media.

To create capture media

1. In the Configuration Manager console, click **Software Library**.
2. In the **Software Library** workspace, expand **Operating Systems**, and then click **Task Sequences**.
3. On the **Home** tab, in the **Create** group, click **Create Task Sequence Media** to start the Create Task Sequence Media Wizard.
4. On the **Select Media Type** page, select **Capture media**, and then click **Next**.
5. On the **Media Type** page, specify whether the media is a flash drive or a CD/DVD set, and then click configure the following:

   - If you select **USB flash drive**, specify the drive where you want to store the content.
   - If you select **CD/DVD set**, specify the capacity of the media and the name and path of the output files. The wizard writes the output files to this location. For example: **\\servername\folder\outputfile.iso**

     If the capacity of the media is too small to store the entire content, multiple files are created and you must store the content on multiple CDs or DVDs. When multiple media is required, Configuration Manager adds a sequence number to the name of each output file that it creates. In addition, if you deploy an application along with the operating system and the application cannot fit on a single media, Configuration Manager stores the application across multiple media. When the stand-alone media is run, Configuration Manager prompts the user for the next media where the application is stored.

     > **IMPORTANT**
     > If you select an existing .iso image, the Task Sequence Media Wizard deletes that image from the drive or share as soon as you proceed to the next page of the wizard. The existing image is deleted, even if you then cancel the wizard.

   Click **Next**.
6. On the **Boot image** page, specify the following information, and then click **Next**.

   > **IMPORTANT**
   > The architecture of the boot image that you specify must be appropriate for the architecture of the reference computer. For example, an x64 reference computer can boot and run an x86 or x64 boot image. However, an x86 reference computer can boot and run only an x86 boot image.

   - In the **Boot image** box, specify the boot image to start the reference computer.

- In the **Distribution point** box, specify the distribution point where the boot image resides. The wizard retrieves the boot image from the distribution point and writes it to the media.

  > **NOTE**
  >
  > You must have Read access rights to the content library on the distribution point.

7. Complete the wizard.